

Communications  
in Computer and Information Science 128

Ana Fred Jan L.G. Dietz Kecheng Liu  
Joaquim Filipe (Eds.)

# Knowledge Discovery, Knowledge Engineering and Knowledge Management

First International Joint Conference, IC3K 2009  
Funchal, Madeira, Portugal, October 6-8, 2009  
Revised Selected Papers

## Volume Editors

Ana Fred  
IST - Technical University of Lisbon  
Av. Rovisco Pais, 1, 1049-001 Lisboa, Portugal  
E-mail: afred@lx.it.pt

Jan L.G. Dietz  
Delft University of Technology  
Mekelweg 4, 2628 CD Delft, The Netherlands  
E-mail: j.l.g.dietz@tudelft.nl

Kecheng Liu  
University of Reading, Informatics Research Centre, Henley Business School  
Reading, RG6 6UD, UK  
E-mail: k.liu@henley.reading.ac.uk

Joaquim Filipe  
Polytechnic Institute of Setúbal  
EST Campus, Estefanilha, 2910-761 Setúbal, Portugal  
E-mail: joaquim.filipe@estsetubal.ips.pt

ISSN 1865-0929  
ISBN 978-3-642-19031-5  
DOI 10.1007/978-3-642-19032-2  
Springer Heidelberg Dordrecht London New York

e-ISSN 1865-0937  
e-ISBN 978-3-642-19032-2

Library of Congress Control Number: 20111920573

CR Subject Classification (1998): H.4, H.3, C.2, H.5, D.2, J.1

© Springer-Verlag Berlin Heidelberg 2011

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

*Typesetting:* Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media ([www.springer.com](http://www.springer.com))

# Preface

The present book includes extended and revised versions of a set of selected papers from the First International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2009), held in Funchal - Madeira, Portugal, during October 6–8, 2009. IC3K was organized by the Institute for Systems and Technologies of Information Control and Communication (INSTICC) in cooperation with ACM SIGMIS.

The purpose of IC3K is to bring together researchers, engineers and practitioners in the areas of knowledge discovery, knowledge engineering and knowledge management, fostering scientific and technical advances in these areas.

IC3K is composed of three concurrent and co-located conferences, each specialized in at least one of the aforementioned main knowledge areas, namely:

- KDIR (International Conference on Knowledge Discovery and Information Retrieval). Knowledge discovery is an interdisciplinary area focusing on methodologies for identifying valid, novel, potentially useful and meaningful patterns from data, often based on underlying large data sets. A major aspect of knowledge discovery is data mining, i.e., applying data analysis and discovery algorithms that produce a particular enumeration of patterns (or models) over the data. Knowledge discovery also includes the evaluation of patterns and identification of which add to knowledge. Information retrieval (IR) is concerned with gathering relevant information from unstructured and semantically fuzzy data in texts and other media, searching for information within documents and for metadata about documents, as well as searching relational databases and the Web. Information retrieval can be combined with knowledge discovery to create software tools that empower users of decision support systems to better understand and use the knowledge underlying large data sets.
- KEOD (International Conference on Knowledge Engineering and Ontology Development). Knowledge engineering (KE) refers to all technical, scientific and social aspects involved in building, maintaining and using knowledge-based systems. KE is a multidisciplinary field, bringing in concepts and methods from several computer science domains such as artificial intelligence, databases, expert systems, decision support systems and geographic information systems. Currently, KE is strongly related to the construction of shared knowledge bases or conceptual frameworks, often designated as ontologies. Ontology development aims at building reusable semantic structures that can be informal vocabularies, catalogs, glossaries as well as more complex finite formal structures representing the entities within a domain and the relationships between those entities. A wide range of applications is emerging, especially given the current Web emphasis, including library science, ontology-enhanced search, e-commerce and configuration.

- KMIS (International Conference on Knowledge Management and Information Sharing). Knowledge management (KM) is a discipline concerned with the analysis and technical support of practices used in an organization to identify, create, represent, distribute and enable the adoption and leveraging of good practices embedded in collaborative settings and, in particular, in organizational processes. Effective knowledge management is an increasingly important source of competitive advantage, and a key to the success of contemporary organizations, bolstering the collective expertise of its employees and partners. Information sharing (IS) is a term used for a long time in the information technology (IT) lexicon, related to data exchange, communication protocols and technological infrastructures.

IC3K received 369 paper submissions from 61 countries in all continents. In all, 32 papers were published and presented as full papers, i.e., completed work, 86 papers reflecting work-in-progress or position papers were accepted for short presentation, and another 62 contributions were accepted for poster presentation. These numbers, leading to a “full-paper” acceptance ratio of 8.7% and a total oral paper presentations acceptance ratio of 32%, show the intention of preserving a high-quality forum for the next editions of this conference. This book includes revised and extended versions of a strict selection of the best papers presented at the conference.

On behalf of the conference Organizing Committee, we would like to thank all participants. First of all to the authors, whose quality work is the essence of the conference, and to the members of the Program Committee, who helped us with their expertise and diligence in reviewing the papers. As we all know, producing a conference requires the effort of many individuals. We wish to thank also all the members of our Organizing Committee, whose work and commitment were invaluable.

July 2010

Ana Fred  
Jan L. G. Dietz  
Kecheng Liu  
Joaquim Filipe

# Organization

## Conference Chair

Joaquim Filipe

Polytechnic Institute of Setúbal /  
INSTICC, Portugal

## Program Co-chairs

Jan L.G. Dietz

Delft University of Technology,  
The Netherlands (KEOD)

Ana Fred

Technical University of Lisbon / IT,  
Portugal (KDIR)

Kecheng Liu

University of Reading, UK (KMIS)

## Organizing Committee

Patrícia Alves

INSTICC, Portugal

Sérgio Brissos

INSTICC, Portugal

Helder Coelhas

INSTICC, Portugal

Vera Coelho

INSTICC, Portugal

Andreia Costa

INSTICC, Portugal

Bruno Encarnação

INSTICC, Portugal

Carla Mota

INSTICC, Portugal

Vitor Pedrosa

INSTICC, Portugal

José Varela

INSTICC, Portugal

Pedro Varela

INSTICC, Portugal

## KDIR Program Committee

Muhammad Abulaish, India

Shu-Ching Chen, USA

Hisham Al-Mubaid, USA

Manolis Christodoulakis, UK

Aijun An, Canada

Dejing Dou, USA

Eva Armengol, Spain

Deniz Erdogmus, USA

Pedro Ballester, UK

Daan Fierens, Belgium

Shlomo Berkovsky, Australia

Ana Fred, Portugal

Isabelle Bichindaritz, USA

Dariusz Frejlichowski, Poland

Florian Boudin, Canada

Benjamin Fung, Canada

Ricardo Campos, Portugal

Vasco Furtado, Brazil

Longbing Cao, Australia

Qigang Gao, Canada

Keith C.C. Chan, Hong Kong

Gautam Garai, India

## VIII Organization

Nazli Goharian, USA  
Jianchao Han, USA  
Haibo He, USA  
Kaizhu Huang, UK  
Yo-Ping Huang, Taiwan  
Samuel Ieong, USA  
Beatriz de la Iglesia, UK  
Szymon Jaroszewicz, Poland  
Rajkumar Kannan, India  
Mehmed Kantardzic, USA  
George Karypis, USA  
Wai Lam, Hong Kong  
Carson K. Leung, Canada  
Changqing Li, USA  
Chun Hung Li, Hong Kong  
Kin Fun Li, Canada  
Tao Li, USA  
Wenyuan Li, USA  
Xiao-Lin Li, China  
Jun Liu, UK  
Paul McNamee, USA  
Rosa Meo, Italy  
Pierre Morizet-Mahoudeaux, France  
Panos Pardalos, USA  
Cheong Hee Park, Korea, Republic of  
Yonghong Peng, UK  
Yanjun Qi, USA  
Zbigniew W. Ras, USA  
Seungmin Rho, USA  
Jia Rong, Australia  
Arun Ross, USA  
Dou Shen, USA

Qiang Shen, UK  
Fabricio Silva, Portugal  
Sergej Sizov, Germany  
Andrzej Skowron, Poland  
Dominik Slezak, Poland  
Nuanwan Soonthornphisaj, Thailand  
Jan Struyf, Belgium  
Marcin Sydow, Poland  
Kay Chen Tan, Singapore  
Ying Tan, China  
Jie Tang, China  
Christos Tjortjis, Greece  
Kar Ann Toh, Korea, Republic of  
Vincent Shin-Mu Tseng, Taiwan  
Celine Vens, Belgium  
Dianhui Wang, Australia  
Hui Wang, UK  
Jeen-Shing Wang, Taiwan  
Zuobing Xu, USA  
Kiduk Yang, USA  
JingTao Yao, Canada  
Yiyu Yao, Canada  
Sule Yildirim, Norway  
Kai Yu, USA  
Xiao-Jun Zeng, UK  
Chengcui Zhang, USA  
Daoqiang Zhang, China  
Min-Ling Zhang, China  
Zhongfei (Mark) Zhang, USA  
Jing Zhou, USA

## KDIR Auxiliary Reviewers

Dino Ienco, Italy  
Yiannis Kanellopoulos,  
The Netherlands

Ruggero Pensa, Italy  
Elena Roglia, Italy

## KEOD Program Committee

Khurshid Ahmad, Ireland	Eyke Hüllermeier, Germany
Yuan An, USA	Masahiro Inuiguchi, Japan
Sören Auer, Germany	Mustafa Jarar, Cyprus
Jean-Paul Barthes, France	C. Maria Keet, Italy
Sonia Bergamaschi, Italy	Tetsuo Kinoshita, Japan
Patrick Brezillon, France	Pavel Kordik, Czech Republic
Silvana Castano, Italy	Chang-Shing Lee, Taiwan
Yixin Chen, USA	Ming Li, China
Wichian Chutimaskul, Thailand	Xiao-Lin Li, China
Nigel Collier, Japan	Weiru Liu, UK
Juergen Dix, Germany	Ralf Möller, Germany
Peter Eklund, Australia	Keiichi Nakata, UK
Dieter A. Fensel, Austria	Adrian Paschke, Germany
Johannes Fuernkranz, Germany	Mihail Popescu, USA
Serge Garlatti, France	Rong Qu, UK
Arnulfo Alanis Garza, Mexico	Seungmin Rho, USA
Paolo Giorgini, Italy	Kiril Simov, Bulgaria
Guido Governatori, Australia	Heiner Stuckenschmidt, Germany
Sergio Greco, Italy	Domenico Talia, Italy
Volker Haarslev, Canada	Dongming Wang, France
Ahmed Hambaba, USA	Junjie Wu, China
Soonhung Han, Korea, Republic of	Slawomir Zadrozny, Poland
Stijn Heymans, Austria	Min-Ling Zhang, China
Angus F. M. Huang, Taiwan	Anna V. Zhdanova, Austria

## KEOD Auxiliary Reviewers

Joonmyun Cho, Korea, Republic of	Heiko Paulheim, Germany
Carmela Comito, Italy	Massimo Ruffolo, Italy
Sebastian Hellmann, Germany	Tae Sul Seo, Korea, Republic of
Frank Loebe, Germany	Gaia Varese, Italy
Stefano Montanelli, Italy	Jeongsam Yang, Korea, Republic of

## KMIS Program Committee

Bernd Amann, France	Dominique Decouchant, France
Aurelie Aurilla Bechina Arnzten, Norway	Asuman Dogac, Turkey
Eva Blomqvist, Sweden	Mariagrazia Dotoli, Italy
Ettore Bolisani, Italy	Joan-Francesc Fondevila-Gascón, Spain
Dickson K.W. Chiu, Hong Kong	Marcus Fontoura, USA

Song Han, Australia	Jean-Henry Morin, Switzerland
Daqing He, USA	Mirella M. Moro, Brazil
Jan Hidders, The Netherlands	Fei Nan, USA
Polly Huang, Taiwan	Antonella Poggi, Italy
Jacky Keung, Australia	Ian Ruthven, UK
Brian Kirkegaard, Denmark	Hiroyuki Tarumi, Japan
Dan Kirsch, USA	Theodore Trafalis, USA
Birger Larsen, Denmark	Rainer Unland, Germany
Feifei Li, USA	Wendy Hui Wang, USA
Mark Manulis, Germany	Andreas Wombacher, The Netherlands
Paolo Merialdo, Italy	Yuqing Melanie Wu, USA
Wolfgang Minker, Germany	Man Lung Yiu, Denmark
Paolo Missier, UK	Clement T. Yu, USA

## KMIS Auxiliary Reviewers

José Guadalupe Rodríguez García, Mexico	Wenjing Ma, USA
Guillermo Morales Luna, Mexico	

## Invited Speakers

Andreas Dengel, German Research Center for Artificial Intelligence (DFKI GmbH), Germany
Madjid Fathi, University of Siegen, Germany
David Jensen, University of Massachusetts Amherst, USA
Jan Dietz, Delft University of Technology, The Netherlands
Steffen Staab, University of Koblenz-Landau, Germany
Justin Zhan, Carnegie Mellon University, USA

# Table of Contents

## Invited Papers

Helping People Remember: Coactive Assistance for Personal Information Management on a Semantic Desktop .....	3
<i>Andreas Dengel and Benjamin Adrian</i>	
Modeling Uncertainties in Advanced Knowledge Management .....	17
<i>Madjid Fathi and Alexander Holland</i>	
Authentication Using Multi-level Social Networks .....	35
<i>Justin Zhan and Xing Fang</i>	

## Part I: Knowledge Discovery and Information Retrieval

Extracting Relationship Associations from Semantic Graphs in Life Sciences .....	53
<i>Weisen Guo and Steven B. Kraines</i>	
Sequential Supervised Learning for Hypernym Discovery from Wikipedia .....	68
<i>Berenike Litz, Hagen Langer, and Rainer Malaka</i>	
Evolving Estimators for Software Project Development .....	81
<i>Athanasis Tsakonas and Georgios Dounias</i>	
Extracting and Rendering Representative Sequences .....	94
<i>Alexis Gabadinho, Gilbert Ritschard, Matthias Studer, and Nicolas S. Müller</i>	
Unsupervised Quadratic Discriminant Embeddings Using Gaussian Mixture Models .....	107
<i>Eniko Szekely, Eric Bruno, and Stephane Marchand-Maillet</i>	
How to Rank Terminology Extracted by EXTERLOG .....	121
<i>Hassan Saneifar, Stéphane Bonniol, Anne Laurent, Pascal Poncelet, and Mathieu Roche</i>	
Average Cluster Consistency for Cluster Ensemble Selection .....	133
<i>F. Jorge F. Duarte, João M.M. Duarte, Ana L.N. Fred, and M. Fátima C. Rodrigues</i>	

## Part II: Knowledge Engineering and Ontology Development

Cross-lingual Evaluation of Ontologies with Rudify .....	151
<i>Amanda Hicks and Axel Herold</i>	
Towards a Formalization of Ontology Relations in the Context of Ontology Repositories .....	164
<i>Carlo Allocca, Mathieu d'Aquin, and Enrico Motta</i>	
ARAGOG Semantic Search Engine: Working, Implementation and Comparison with Keyword-based Search Engines.....	177
<i>Harsh Mittal, Jaspreet Singh, and Jitesh Sachdeva</i>	
Cooperative WordNet Editor for Lexical Semantic Acquisition .....	187
<i>Julian Szymański</i>	
Analogical Cliques in Ontology Construction .....	197
<i>Guofu Li and Tony Veale</i>	
Detection and Transformation of Ontology Patterns .....	210
<i>Ondřej Šváb-Zamazal, Vojtěch Svátek, François Scharffe, and Jérôme David</i>	
Islands and Query Answering for Alchi-ontologies .....	224
<i>Sebastian Wandelt and Ralf Möller</i>	
Ontology Co-construction with an Adaptive Multi-Agent System: Principles and Case-Study .....	237
<i>Zied Sellami, Valérie Camps, Nathalie Aussenac-Gilles, and Sylvain Rougemaille</i>	
A Methodology for Knowledge Acquisition in Consumer-Oriented Healthcare .....	249
<i>Elena Cardillo, Andrei Tamilin, and Luciano Serafini</i>	
Combining Statistical and Symbolic Reasoning for Active Scene Categorization .....	262
<i>Thomas Reineking, Niclas Schult, and Joana Hois</i>	
A Semi-automatic System for Knowledge Base Population.....	276
<i>Jade Goldstein-Stewart and Ransom K. Winder</i>	

## Part III: Knowledge Management and Information Sharing

Enterprise Wikis - Types of Use, Benefits and Obstacles: A Multiple-Case Study .....	297
<i>Alexander Stocker and Klaus Tochtermann</i>	

A Knowledge Management System and Social Networking Service to connect Communities of Practice.....	310
<i>Élise Lavoué</i>	
Design Issues for an Extensible CMS-Based Document Management System .....	323
<i>João de Sousa Saraiva and Alberto Rodrigues da Silva</i>	
CrimeFighter: A Toolbox for Counterterrorism.....	337
<i>Uffe Kock Wiil, Nasrullah Memon, and Jolanta Gniadek</i>	
Functional Analysis of Enterprise 2.0 Tools: A Services Catalog .....	351
<i>Thomas Büchner, Florian Matthes, and Christian Neubert</i>	
Scenario Process as a Community for Organizational Knowledge Creation and Sharing .....	364
<i>Hannu Kivijärvi, Kalle A. Piirainen, and Markku Tuominen</i>	
Neural Networks Aided Automatic Keywords Selection .....	377
<i>Błażej Zyglarski and Piotr Bala</i>	
Value Knowledge Management for Multi-party Conflicts: An Example of Process Structuring .....	390
<i>Shahidul Hassan and John Rohrbaugh</i>	
Leveraging Organizational Knowledge Management through Corporate Portal .....	399
<i>Kamla Ali Al-Busaidi</i>	
<b>Author Index .....</b>	<b>411</b>

# **Invited Papers**

# Helping People Remember: Coactive Assistance for Personal Information Management on a Semantic Desktop

Andreas Dengel<sup>1,2</sup> and Benjamin Adrian<sup>1,2</sup>

<sup>1</sup> DFKI, Knowledge Management Dept.

Trippstadter Straße 122, 67663 Kaiserslautern, Germany

<sup>2</sup> University of Kaiserslautern, Computer Science Dept., 67663 Kaiserslautern, Germany

{andreas.dengel, benjamin.adrian}@dfki.de

**Abstract.** This paper proposes a co-active information butler giving users assistance on a Semantic Desktop. The information butler is a novel approach for managing documents and information entities on a Semantic Desktop. The Semantic Desktop encourages users to create, label, describe, and interlink information objects (such as emails, files, address book records, etc.) above the level of any desktop application. The proposed information butler recommends users which parts of their information models to take for categorizing and interlinking documents with concepts. Ontology-based document understanding is shown as key technology for implementing such an information butler that automatically recommends concepts of the user's personal information models that have evidences to be relevant for a certain document.

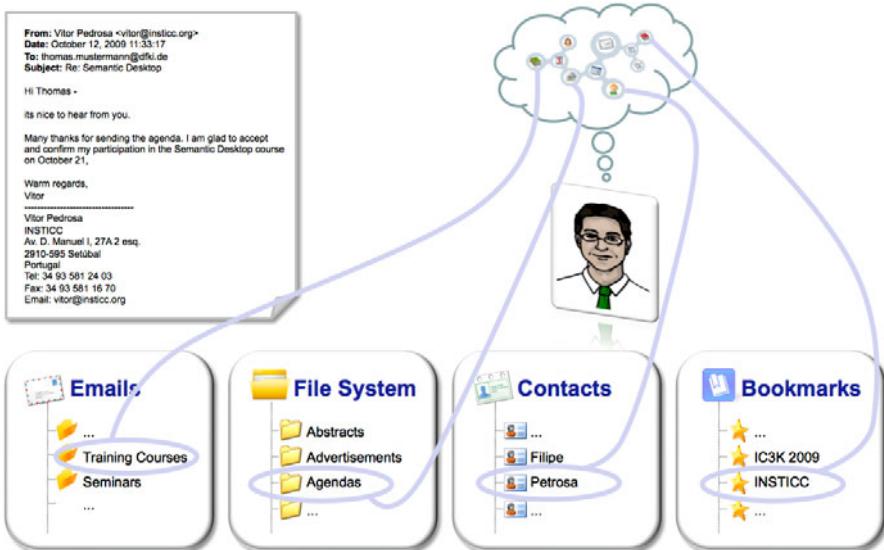
**Keywords:** Semantic desktop, Personal information management, RDFS, Information extraction, Semantic web, Knowledge management.

## 1 Introduction

In order to motivate our approach of a co-active information butler, we will start with a scenario that may be found at many workplaces around the globe where *knowledge workers* act as the mental clue interconnecting complex and heterogeneous information spaces to drive their processes and solve their task without getting real support in personal information management.

Dr. Thomas Mustermann is such a typical but fictive knowledge worker. He is head of Customer Relationship Management at the German Research Center for Artificial Intelligence (DFKI) in Kaiserslautern, Germany. He is one of those guys who are tackled by Constant Multi-Tasking Craziness. Right now he works on about 15 tasks at the same time one of which is the organization of a training course on DFKI's recent development: *The Semantic Desktop*. When his colleague Andreas Dengel came back from IC3K 2009 held on the island of Madeira, Portugal, he told Thomas to invite Vitor Petrosa, another knowledge worker, to the training course.

In order to fulfill his role and to complete all of his tasks, including the one he just received, Thomas has to acquire, organize, maintain, retrieve and use a whole bunch of information items of different quality. This also holds for the before-mentioned



**Fig. 1.** A single email message is related to a variety of information items captured in different applications. Thomas unifies these information bits in his mental model, but keeps it under his hat. The overall scenario stays hidden to desktop application.

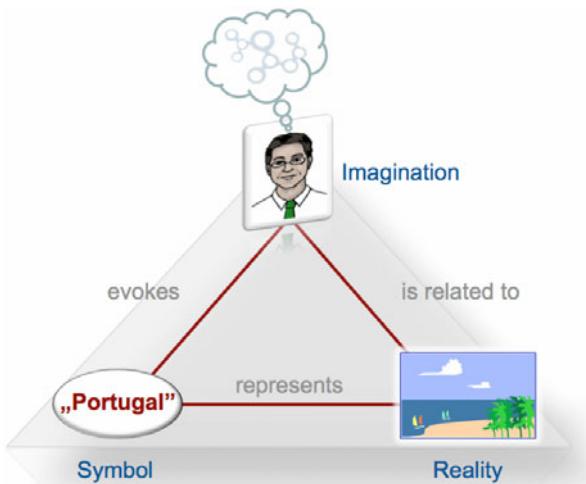
training course, for which he has to establish an agenda, fix a date in his calendar, and organize an appropriate room. Furthermore, he has to prepare the presentation and send out the invitations to potentially interested leads.

However, Thomas' problem, as for many knowledge workers, is the need to know more than he may remember. On Thomas' desktop, he stores around 11,000 files in about 1,200 folders in many applications and formats. All of these folders contain relevant aspects of information that are needed in a specific situation, a running task, a forthcoming appointment, or an arising topic. In order to do his job, Thomas organizes the folders on his desktop applications in a way he may well use them. When sending out the invitation to Vitor, Thomas may employ State-of-the-Art technologies extracting and storing metadata from the email header, namely sender and recipient as well as date and time of the transmission. Furthermore the text in the subject and the body of the message could be examined for categorization and indexing. As a result, Thomas is able to file and to retrieve the information by standard tools.

However, having a closer look, the message sent by Thomas is related to many bits of information captured in various different applications, such as address data of Vitor, the agenda of the event, the time slot entered into Thomas' calendar, the presentation and the training course handouts he already prepared for the event. Unfortunately the ways a computer traditionally stores and organizes information access do not allow representing the inherent relations among the information sources, i.e., relevant information objects are resident at different places and within independent applications. As exemplarily shown in Fig 1:

- Emails are filed in Email folders
- Attachments are put into directory folders of the file system
- The sender of an Email is stored in an independent address repository
- Related Websites about product descriptions or other information are disregarded

As a consequence many of these implicit relationships are lost over time or are only present as tacit knowledge in Thomas' mental models but are not obvious for other users because they are not explicitly represented on Thomas' desktop.



**Fig. 2.** Semiotic Triangle. Here, Thomas has his own, subjective imagination about the real world concept Portugal that comes to his mind when e.g., reading the symbol “Portugal”.

When Vitor's reply would arrive (in reasonable time) in Thomas' inbox, as illustrated by Fig. 1, Thomas may easily relate the content of the answer into context because he has necessary background knowledge. However, there is no way for Thomas to file the email in all of its multi-perspective aspects which would help to retrieve the appropriate contextual categories later on, at any time he needs them. This example peels the limits of today's desktop information management and reveals the big gap between mental information processing and personal information management. So what can we do in order to help knowledge workers like Thomas to do their job better?

Documents are a means for indirect communication. They may precisely describe or vaguely denote facts that can only be understood in the context of well-known categories of events, locations, persons, topics or concrete instances of them. Drains of thought drive a user's attention and mental models allowing a perspective consideration of content by weighting the aspects and circumstances of a message different. So, while reading people create models of understanding, which are composed of snapshots of the text they read and plausible beliefs created from similar experience in their memory. These mental models sharpen our attention and determine the intent of a message.

Thus, a document is like key, which while reading opens up a system of links to other documents, to similar cases, existing solutions, or known experts. As part of a

process, new facts and relationships captured in the document are learned and complement the existing mental models. This way, Ogden & Richard's [2] famous semiotic triangle of meaning implies that the unit of a message is variable and relative, depending on who reads it at what time and in which context.

As shown in Fig. 2, our environment consists of items, facts and events that are "real" and determine our daily life ("what is going on"). However, for expressing our thoughts, we make use of symbols, signs, or characters that may be understood by others ("what I couch or explicate"). A symbol can be considered as a conceptualization, i.e. it is something that stands to somebody for something in some respect or capacity. While reading text, we imagine putting pieces of content together and create our very individual understanding ("what I mean").

The same happens to Thomas, who, when receiving new information, intuitively relates the contents of a message to his mental models. Although he tried to mirror the concepts on his desktop, the associations is tacit and only happens in his thoughts. So this semiotic relations are implicitly applied to his work context while his brain bundles all relevant aspects of a message revealing associations and activating his mental relationships.

In order to transform a computer into a vivid *information butler*, we have to first consider that the meaning of the Latin word "communicare" is to make common. This implies that communication requires a shared model of understanding that would allow a butler, as proposed by Isaacs and Walendowski [3], to anticipate what we think - a butler being always available, paying attention to the user, and observing her/his action on documents in order to learn conceptual relationships and to propose appropriate concepts in a given context?

For describing how such an information butler works, we like to first give an introduction into the Semantic Desktop metaphor [5] we have developed at DFKI intending to provide an active work support for semantic markup and proactive semantic search services.

## 2 The Semantic Desktop

Like the Semantic Web our approach builds on predication and ontologies to formally represent semantics. While the ontology attempts to give answers to the question: "What is there?", predicates try to answer the question: "What is it to say something about something?". In other words, while a subject is what a statement is about, a predicate is what a statement says about its subject.

In the context of the Semantic Desktop, an ontology as defined by Gruber [1], is an explicit, formal specification of a conceptualization, i.e. a particular vocabulary that can be employed for describing aspects of real domains. To do so, we make use of the enhanced Resource Description Framework (RDF [11]) where meaning is expressed by facts encoded in sets of triples. Triples are like elementary sentences composed of subject, predicate, and object. Subjects, predicates, and objects are given as names for entities, also called resources or nodes. Entities represent something, a contact, an appointment, a website, etc. Names are either literals or Uniform Resource Identifiers (URI [13]), which are global in scope, always referring to the same entity in any RDF document in which they appear. Fig. 3 shows an example of an RDF statement describing that "Thomas has the phone number ...."



**Fig. 3.** RDF represents information as graph in form of triples. Triples consist of subject, predicate, and object values. RDF can be serialized in XML.

The underlying structure of any knowledge represented by such statements can be viewed as a graph (of triples) consisting of nodes (subjects, objects) and labeled directed arcs (predicates) that link pairs of nodes. Following the statement of Tim Berners-Lee, Jim Hendler, and Ora Lassila in their article in the Scientific American [4], the Semantic Desktop is “not a separate” desktop, “but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation.” In other words, it may be seen as a device in which an individual stores all her/his digital information like documents, multimedia, and messages. All information objects are interpreted as Semantic Web resources identified via respective URIs. Since the resources are available as part of an RDF graph, they are accessible and queryable. Meta models are expressed via an ontology, providing the shared vocabulary and forming the semantic glue to interconnect information from various applications and allowing the user to share her/his resources with others via Semantic Web protocols. This way, we have designed and developed the Semantic Desktop is an enlarged supplement to the user's memory [5].

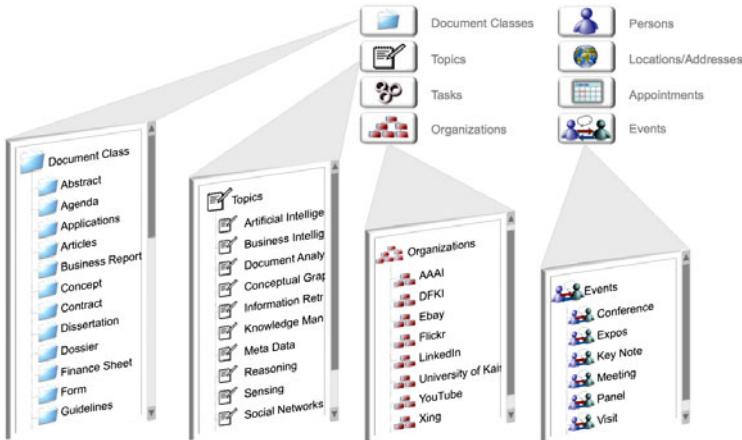
An information butler is observing the user's activities, analyzing the documents under consideration and assisting the user to relate and access all relevant information across applications. In order to do so, a user may employ a system of categories as exemplary shown in Fig. 4 for formally describing a vocabulary of her/his work context as well as facts about it.

Since Thomas, our fictive knowledge worker, already uses the ontological categories of the Semantic Desktop, he was able to create an application-independent Personal Information Model (PIMO) [6] on all his resources.

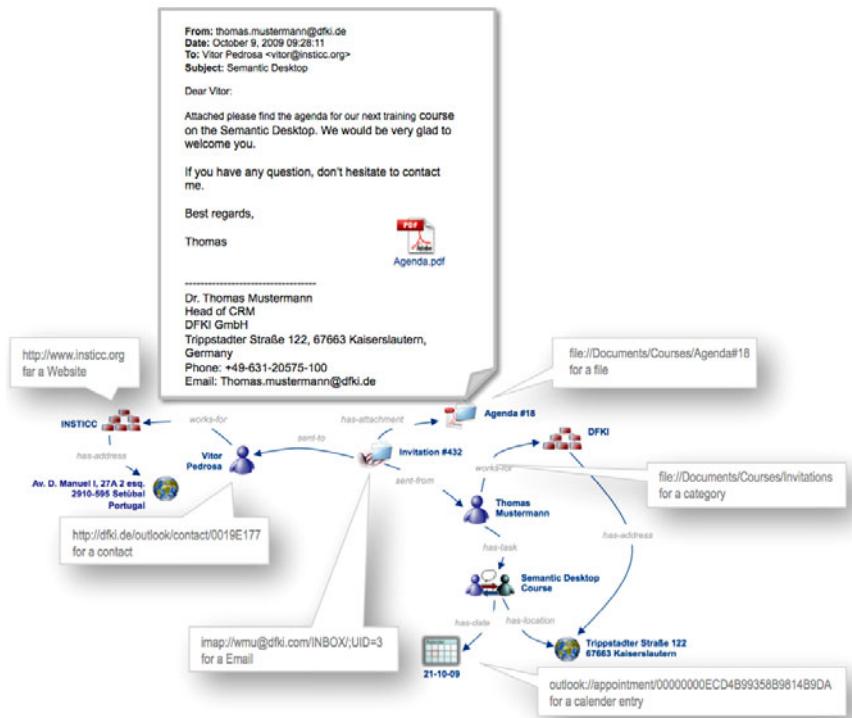
The email, for example, Thomas sent to Vitor for inviting him is already part of his PIMO as shown by the triple graph in Fig. 5. Each information item is like a semantic web resource identified by its URI, no matter whether it is a file (folder or document), an email constituent (i.e. message, sender, recipient, and attachment), an address, or calendar entry.

Note that such a formal representation of mental concepts via a triple graph leads to a multi-perspective organization of content and for this reason necessarily to a “dematerialization” of traditional filing concepts we discussed at the beginning of this paper and consequently overcomes the inherent problems of separate desktop applications. Moreover, an information butler committing to the PIMO may use the pre-given vocabulary in a consistent way, and thus can exchange information and reason about entities in the modeled domain.

The information butler bases his recommendations on existing information inside the PIMO. It annotates new documents by known categories or other PIMO instances. Besides just recognizing known entities in the text, the butler may infer facts that are not explicitly part of a document.



**Fig. 4.** Thomas' Semantic Desktop allows the maintenance of his personal information model (PIMO). The PIMO contains Thomas' vocabularies, his personal categorization scheme and his information entities as manifestations of Thomas' mental concepts.



**Fig. 5.** The graph of information items represents the part of Thomas' PIMO that is relevant for the email Thomas is currently writing

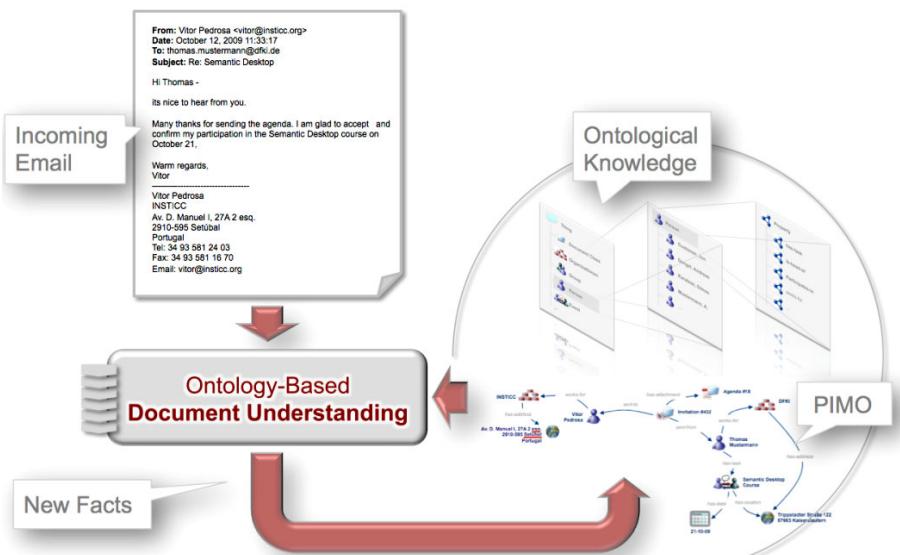
This way, Thomas may see what the butler can do and what he does by employing the annotation proposals in order to grow his PIMO by a synergetic manual and semi-automatic adding. As expressed by [3] such complementary behavior may lead to a synergetic collaboration in socio-technical setting. User and information butler are learning from each other how to improve productivity without noticing the interaction.

### 3 Ontology-Based Document Understanding

For anticipation, the information butler employs an ontology-based document understanding system developed at DFKI called SCOOBIE (ServCe fOr Ontology-Based Information Extraction). SCOOBIE stepwise transforms document text content into formal facts inside the PIMO as a supplement for Thomas' mental models. Facts may be triples of two kinds:

- An attribute like statement literal values called datatype properties (e.g., family names of Thomas address book contacts), or
- A relationship between PIMO concepts called object properties (e.g., Thomas is employee of DFKI).

Extracted facts are either known facts, whose triples exist in Thomas PIMO, or yet unknown facts, which means they do not exist in Thomas PIMO, yet, because Thomas did not know or had not the time to create them explicitly.

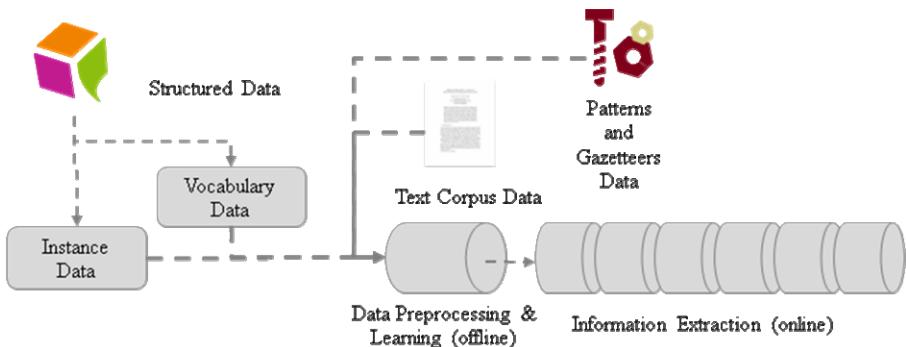


**Fig. 6.** Ontology-based document understanding is capable for extracting information entities from documents that are represented in the ontology's vocabulary and classification scheme. Vice versa, it may annotate document passages with semantic markup, whenever the passage represents a symbolic label for an existing concept inside Thomas' PIMO.

### 3.1 Ontology-Based Information Extraction with SCOOBIE

In SCOOBIE, ontology-based information extraction algorithms incorporate those relevant bits of knowledge from Thomas' PIMO that support extraction tasks inside a pipeline of cascading extraction tasks [7]. Fig. 7 shows the architecture of SCOOBIE. It takes structured information (vocabulary, instances, regular expression, and word lists), and unstructured but domain describing information (a document corpus) as input, preprocesses them offline, and finally uses the resulting models, data and index structures for online information extraction. Conceiving SCOOBIE's extraction pipeline as black box, its mandatory input parameters are:

- SCOOBIE needs a representation of Thomas' PIMO in RDF. This may consist of several vocabularies (e.g., FOAF<sup>1</sup> for expressing information about persons, or GeoNames<sup>2</sup> for expressing information about locations) that Thomas used for describing his personal concepts. The classes (e.g., persons, or companies), datatype properties of these classes (e.g., person's first and last names, or addresses of company headquarters) and object properties between instances of these classes (e.g., persons being employed in companies) define a search space of possible instances and facts that may be retrieved from text.



**Fig. 7.** Architecture of SCOOBIE

Besides vocabularies, SCOOBIE requires Thomas' concepts and work items that are represented as instances inside his PIMO. These instance data is used for creating index structures and other models used for specific extraction tasks. For example, the datatype property values of existing instances are used for identifying these instances in text. Object properties between instances are used for disambiguating instances with similar datatype property values (e.g., persons with identical names).

<sup>1</sup> The Friend-of-a-Friend vocabulary (see <http://xmlns.com/foaf/spec/>) allows representing common properties about persons, their interests and activities, and their relations to companies, projects.

<sup>2</sup> The GeoNames vocabulary (see <http://www.geonames.org/ontology/>) allows representing cities, and countries with properties such as names, local names, altitude and longitude coordinates, etc.

In addition, Thomas also used optional parameters for tweaking the extraction capabilities of his information butler to his personal flavors.

- Thomas passed a corpus of text documents that are already categorized inside his Semantic Desktop. In an automatic training step, SCOOBIE annotates these documents with all matches of datatype property values of existing instances and trains machine-learning models (such as a conditional random field, see [8] for details) for recognizing similar phrases being possible candidates for new datatype property values in these existing or new documents.
- Thomas also passed several word lists (called gazetteers) consisting of city names, country names, company names, and person names. He also added regular expressions about research grant identifiers to a list of regular expressions about common structured values such as email addresses, or dates. In a learning step, the system recognizes matches between gazetteer entries or regular expressions and known datatype property values. These matches are counted to build heuristics about which datatype property to use whenever certain gazetteer entries or regular expressions occur in text.
- Depending on the class of documents (email, slides, or homepage), Thomas used a concrete RDF query expressed in SPARQL in order to explicitly filter relevant types of facts that should be extracted from text.

Besides the queries, these parameters are analyzed during an offline preprocessing and training phase. Results are index structures (e.g., suffix arrays, B\*-trees) and learning models (e.g., Conditional Random Fields, K-Nearest neighbor Classifiers) that can now be used by efficient extraction tasks inside the extraction pipeline:

- **Normalization:** Extracts document metadata and the plain text data from textual or binary file formats. The language of the plain text is detected by applying statistics about n-gram distributions of letters in different languages.
- **Segmentation:** Partitions plain text into segments: paragraphs, sentences, and tokens. With respect to the detected language, a POS tagger tags each token with its part of speech (POS).
- **Symbolization:** Recognizes datatype property values in text. It matches phrases in text and values of datatype properties inside the PIMO. (e.g., assuming the triple  $<...> \text{rdfs:label } \text{"DFKI"}$ , “DFKI” may be recognized as content symbol of type rdfs:label in text).

By applying existing gazetteers and regular expressions, Symbolization also performs Named Entity Recognition and Structured Entity Recognition.

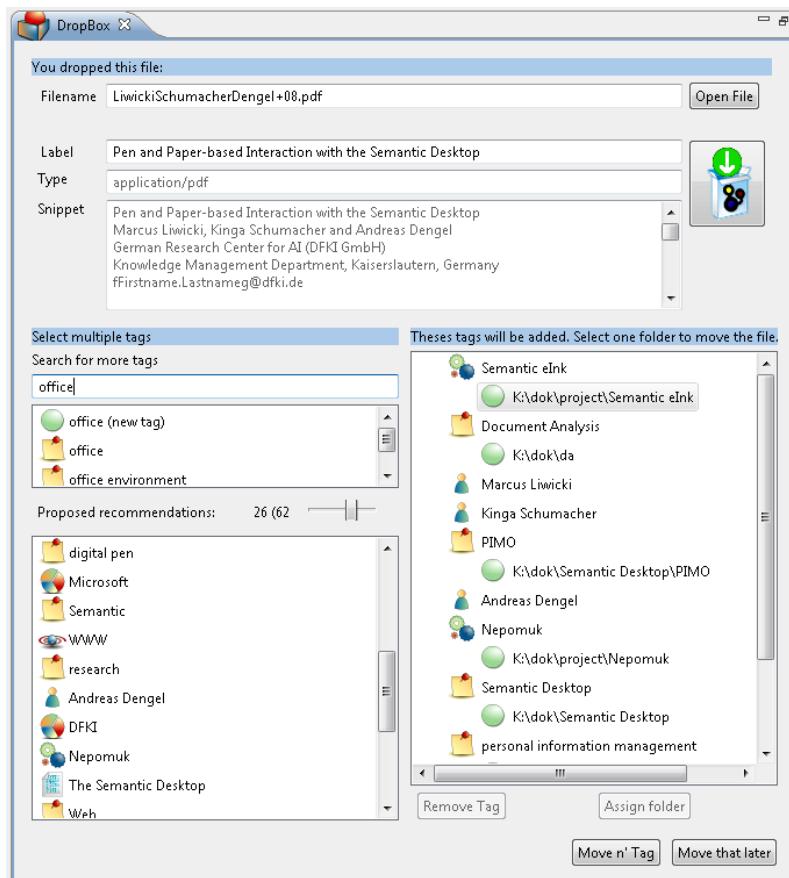
Noun phrase chunking recognizes phrases as candidates for new concept names.

- **Instantiation:** For each recognized datatype property value (e.g., assuming the triple  $(:DFKI \text{rdfs:label } \text{"DFKI"})$ , “DFKI” may be resolved as rdfs:label of instance :DFKI), the instantiation resolves existing instances of a PIMO.

Instance candidate recognition resolves possible candidates for recognized datatype property values.

Here, ambiguities may occur if more than one instance possesses the same datatype property values (e.g., the first names of Thomas Mustermann and Thomas Kieninger). Candidates are disambiguated by counting resolved instances inside the PIMO that are related directly with an object property or indirectly via another instance of the PIMO. As result, the ambiguous instance candidate with the higher count of related and resolved instances is taken.

- **Contextualization:** Extracts facts (RDF triples) about resolved instances. At first, a fact candidate extraction computes all possible facts between resolved instances. Then, a set of fact selectors rates these facts according to heuristics. A known fact selector heightens rates of extracted facts that exist as triples inside the domain model.
- **Population:** Creates scenario graphs in RDF format. Creates scenario contain extracted values, i.e., URIs of resolved instances with those datatype property values that match with text sequences and RDF triples about object properties between these resolved instances. Scenario graphs can be filtered and ordered by confidence values in range between zero and one.



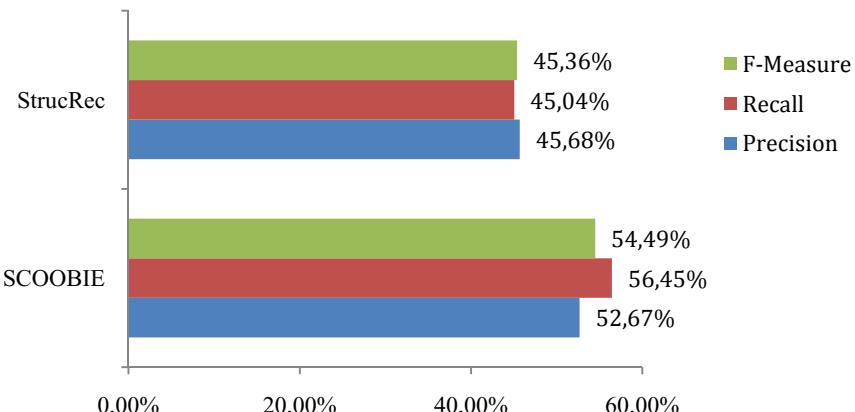
**Fig. 8.** The Nepomuk DropBox application

### 3.2 Recommending PIMO Instances as Semantic Tags for Documents

Thomas uses a simple application implementing the information butler paradigm for categorizing incoming documents with his PIMO concepts. The so-called drop box allows quick and easy document categorization as shown in Fig. 8. The upper part

contains a small document summarization with title, mime type, and text snippet. Left side shows tags recommended by SCOOBIE's information extraction. Users may add additional tags manually. Right side shows instances already connected to this document as tags.

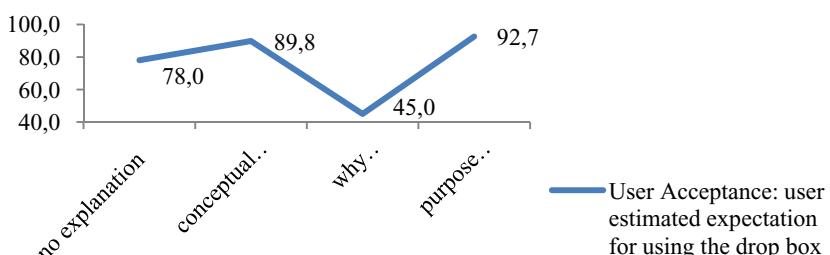
The drop box polls a dedicated folder in Thomas' file system. Whenever Thomas drops a document into this folder, SCOOBIE extracts relevant PIMO instances and presents them as tag recommendations in a minimalistic popup user interface. Now, Thomas can quickly relate instances to this document by double clicking.



**Fig. 9.** Results of a quantitative analysis about tags recommended by SCOOBIE and StrucRec. StrucRec is the standard tag recommender in Nepomuk Semantic Desktop.

## 4 Evaluating the Information Butler

The quality of SCOOBIE's tag recommendations was evaluated in a quantitative evaluation measured with recall and precision ratios [9]. Five users agreed to let us evaluate SCOOBIE's tag recommendations based on their private PIMO data. They used the Nepomuk Social Semantic Desktop [14] implementation. The PIMO models differed in size and connectivity between instances.



**Fig. 10.** Influence of explaining recommended PIMO concepts differently on the users' general expectation about using the system in daily work

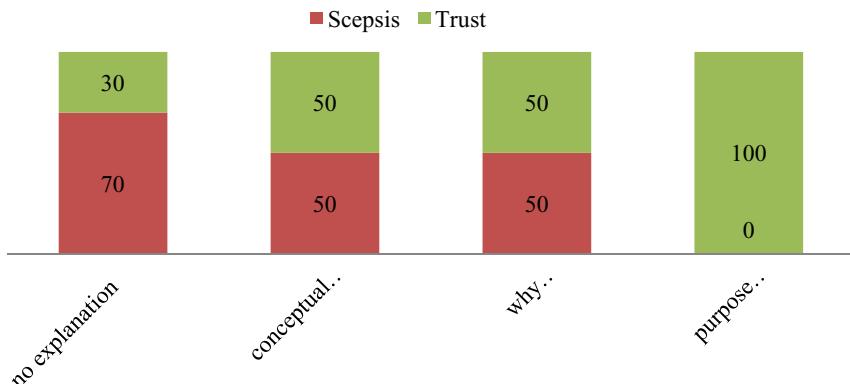
As shown in Fig. 9, SCOOBIE's tag recommendations were rated by users with 56.45% recall and 52.67% precision. These results beat the standard tag recommender in Nepomuk called StrucRec [9]. StrucRec's results were rated with 45.04% recall which is 11.41% less than SCOOBIE and 45.68% precision which is -6.99% less than SCOOBIE.

The analysis revealed that both systems generated better results in recall and precision for those PIMOs whose instances are highly connected.

How users interacted and tended to accept working with such an information butler was evaluated in a more qualitative analysis [10]. Here, the used measure is called *user acceptance*. It rates the estimated expectation of users for using the information butler's (i.e., drop box) concept recommendations about document in their daily work. The evaluation was set up by giving users an assumed PIMO, a certain document with content that relates the PIMO's content, and a set of concepts the information butler recommends as categories for this document.

Users were asked if they would use these kinds of recommendations for categorizing documents within their daily work. They should rate their expectation with a value between 0 and 100%. Users were also asked if they trust in the relevancy of these concepts to the certain document.

In following steps, different kinds of explanations about why the system recommended these concepts were given. After each of these explanations, the users should again decide which instance to accept and tell if they now would more or less likely use the recommender with this kind of explanation. As shown in Fig. 10, initially the average expected user acceptance for a tag recommender was observed to be very high and set to 78%. Additional conceptual explanations showing relevant parts of the PIMO about a certain concept increased the user acceptance demonstrably (+11.88%). In contrast, why-explanations about intermediate results of SCOOBIE's extraction pipeline decreased user acceptance dramatically (-44.82%). The users did not want to know how SCOOBIE generates a certain recommendation. Finally, purpose explanations giving insight of existing SPARQL<sup>3</sup> queries that specify filters about what kind of concepts to recommend raised the user acceptance (+47.67%).



**Fig. 11.** Influence of explaining recommended PIMO concepts differently on the general trust in the relevancy of these PIMO concepts related to the current document

<sup>3</sup> <http://www.w3.org/TR/rdf-sparql-query/>

Fig. 11 shows that giving without any explanation about recommended concepts users trusted in 30% of the recommendations. Given conceptual or why-explanations raised the trust rate up to 50%. Purpose explanations showing existing filters finally raised the trust rate to 100%.

## 5 Conclusions and Outlook

In this paper we presented a novel approach of personal information management on a Semantic Desktop. The notion of a coactive information butler was introduced and explained by pointing out its advantages in a fictive scenario of Thomas Mustermann who organizes a training course about the Semantic Desktop. The information butler helps Thomas in linking documents such as emails to relevant parts of his personal information model (PIMO) that is his Semantic Desktop solution of explicating the hidden conceptual scenarios inside his mental model.

We presented a prototypical application of such an information butler called DropBox. For each document dropped into this box, the butler recommends relevant concepts of Thomas PIMO for categorizing the document within his PIMO. We give details on ontology-based document understanding techniques as an approach for implementing a DropBox and present the SCOOBIE system with its information extraction pipeline that finally generates concept recommendations by considering the document content and current information inside a PIMO.

In a quantitative evaluation the SCOOBIE approach produced better recommendations than the existing concept recommender of the Nepomuk Semantic Desktop called StrucRec.

A qualitative evaluation revealed that users tend to accept and work in collaboration with such an information butler if it provides comprehensible explanations about its recommended concepts.

This work figured out that using a PIMO for explicating and maintaining mental models on a desktop allows to develop sophisticated and user adaptable assistance systems. These systems, if presented as well explaining coactive information butlers, have the potential to increase efficiency of knowledge workers.

This work was funded by the BMBF project Perspecting (Grant 01IW08002).

## References

1. Gruber, T.: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition* 5(2), 199–220
2. Ogden, C.K., Richards, I.A.: *The Meaning of Meaning: A Study of the Influence of Language Upon Thought and of the Science of Symbolism*. Routledge & Kegan Paul, London (1923)
3. Isaacs, E., Walendowski, A.: Designing from Both Sides of the Screen: How Designers and Engineers Can Collaborate to Build Cooperative Technology. New Riders Publ., Indianapolis (2001)
4. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *Scientific American* (2001)
5. Sauermann, L., Bernardi, A., Dengel, A.: Overview and Outlook on the Semantic Desktop. In: Proceedings ISWC, 6<sup>th</sup> Int. Semantic Web Conference, Galway, Ireland, pp. 1–19 (2005)

6. Sauermann, L., van Elst, L., Dengel, A.: PIMO - a Framework for Representing Personal Information Models. In: Proc. I-Semantics 2007, Graz, Austria, pp. 270–277 (2007)
7. Adrian, B., Hees, J., van Elst, L., Dengel, A.: iDocument: Using Ontologies for Extracting and Annotating Information from Unstructured Text. In: Mertsching, B., Hund, M., Aziz, Z. (eds.) KI 2009. LNCS, vol. 5803, pp. 249–256. Springer, Heidelberg (2009)
8. Adrian, B.: Incorporating ontological background knowledge into Information Extraction. In: Maynard, D. (ed.), ISWC 2009: Doctoral Consortium, Washington DC, US (2009)
9. Adrian, B., Klinkigt, M., Maus, H., Dengel, A.: Using iDocument for Document Categorization in Nepomuk Social Semantic Desktop. In: Proceedings of I-KNOW 2009 and I-SEMANTICS 2009, 5th International Conference on Semantic Systems (iSemantics 2009), September 2-4, pp. 638–643. Verlag der Technischen Universität Graz, Graz (2009) ISBN 978-3-85125-060-2
10. Adrian, B., Forcher, B., Roth-Berghofer, T., Dengel, A.: Explaining Ontology-Based Information Extraction in the NEPOMUK Semantic Desktop. In: Workshop 10@IJCAI-09: Explanation-aware Computing (ExaCt 2009), Int. Joint Conference on Artificial Intelligence (IJCAI 2009), Pasadena, California, United States, pp. 94–101. AAAI, Menlo Park (2009)
11. W3C: RDF Primer. W3C Recommendation (2004),  
<http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>
12. W3C: SPARQL Query language for RDF. W3C Recommendation (2008),  
<http://www.w3.org/TR/rdf-sparql-query/>
13. Berners-Lee T., Fielding R., Masinter L.: RFC 2396 - Uniform Resource Identifiers (URI): Generic Syntax, IETF (August 1998),  
<http://www.isi.edu/in-notes/rfc2396.txt>
14. Bernardi, A., Decker, S., van Elst, L., Grimnes, G., Groza, T., Handschuh, S., Jazayeri, M., Mesnage, C., Moeller, K., Reif, G., Sintek, M.: The Social Semantic Desktop: A New Paradigm Towards Deploying the Semantic Web on the Desktop. In: Semantic Web Engineering in the Knowledge Society. IGI Global (October 2008)

# Modeling Uncertainties in Advanced Knowledge Management

Madjid Fathi and Alexander Holland

Institute of Knowledge Based Systems & Knowledge Management  
University of Siegen, Siegen, Germany

**Abstract.** Handling and managing knowledge is a difficult and complex enterprise. a wide range of advanced technologies have to be invoked in providing assistance for knowledge requirements ranging from acquisition, modeling, (re)using, retrieving, publishing and maintaining of knowledge. Knowledge engineers swap ideas, communicate, plan, act, or reason often in situations where facts are unknown and the underlying natural language is uncertain or vague. They do not have access to the complete environment to evaluate each situation. Also conditions are unknown, incomplete or only crudely summarized. In this paper, knowledge representation applications as formal graphical language are examined in detail. exempli gratia, bayesian networks are graphical models to represent knowledge under conditions of uncertainty. This network type model the quantitative strength of the connections between variables allowing probabilistic beliefs about them to be updated automatically as new information becomes available. Applications in various fields like mechanical engineering are exemplified.

**Keywords:** Product use information, Product lifecycle management, Knowledge representation, Uncertainty management, Artificial intelligence.

## 1 Introduction

Knowledge management originates at least on three roots. At first, suppliers of information technology and academics in this field have developed opportunities of supporting knowledge tasks by knowledge-based systems, artificial intelligence and web-based applications. Secondly, organization and human relations professionals have recognized the need for using the opportunities of an increasingly highly educated work force in modern societies. At last, strategic management has recognized that the optimal use of intellectual capabilities may be the best source for sustaining competitiveness in the global economy. Knowledge engineers exchange experiences where issues are unknown or only partially accessible. Reasoning in realistic domains requires some kind of simplification or adaption to deal with exceptions or to increase the degree of belief in decision making situations. Reasoning under uncertainty is interchangeably associated with the handling of uncertain knowledge. The probability theory has an outstanding position and serves as basis for human's behavior in decision situations. Uncertainty solutions must solve occurring questions in how to represent uncertain data, how to combine pieces of uncertain data and how to draw inference with uncertain data. Techniques are

mainly separated in quantitative and symbolic approaches. Quantitative approaches using certainty factors, fuzzy logic, belief functions, probabilities and possibilities. Symbolic approaches dealing for instance with circumscriptions or default logic. Mechanical engineering provides an important application field wherein knowledge representation techniques are deployed. Design Simulation and Design for X Methods and tools support the anticipation of product behavior during its use [1,2]. Unfortunately, the real conditions of the product use and environmental use conditions differ from the design assumptions. Today the acquisition, aggregation and analysis of product field data for design purposes are fairly difficult due to different reasons.

First, within the current producer-customer business models the product suppliers do not have any access to the customer's product environment. Second, sensors embedded within products for monitoring product use parameters like loads and environmental data are rarely used due to their high price and their large size. Third, an integrated theoretical framework for filtering, aggregating and analyzing field data for design feedback as well as appropriate IT infrastructures are not available.

The emerging shift within manufacturing companies from selling products to offering customer specific product service systems (IPS<sup>2</sup>) will expand the responsibility of producers to the whole product lifecycle [3] and will facilitate an easier access to product use information. Furthermore, the progress in the miniaturization of embedded micro sensors, their price reduction as well as advances in the information technology will allow an easier capturing and processing of product use information as feedback for the development of improved products.

In this changed industrial and technological environment the project described in this paper aims at the development of a new solution for the acquisition, aggregation and analysis of product use information. This solution is based upon knowledge-based methods like Bayesian network inference and is integrated in an extended Product Lifecycle Management (PLM [4]) solution. Within the scope of a feedback cycle, product use information and deduced knowledge from previous product generations can be incorporated into the development of subsequent product generations in a target-oriented fashion and can thus provide faster product improvements, lower development costs, increased product quality and lower maintenance expenses for the use phase.

## 2 Uncertainty Management

Humans plan, act, or reason in situations where facts are often unknown or uncertain [5]. They do not have access to the complete environment to evaluate each scenario. Conditions and facts are unknown, incomplete or only crudely summarized. Supposedly, for example, a situation in which a car driver wants to drive a colleague to the next airport to catch a flight and considering the time leaving from home before the flight. In the case of planning 90 minutes leaving time before the estimated time of departure and a distance of 15 kilometers to the airport and an average rate of 50 kilometers/hour the car driver can only act under uncertainty. He is not able to follow if he gets into an accident, if a flat tire occurs or if he is caught up in a traffic jam during the driving time. But the handling of uncertain knowledge is calculated and well considered to avoid unproductive waiting time at the airport or speeding tickets

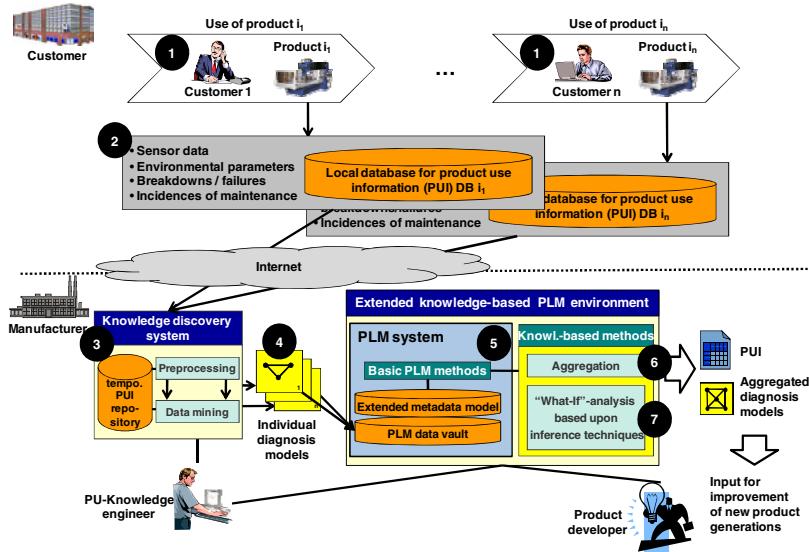
along the driving way. Another situation is the codification of knowledge available in rules such as “much alcohol suggests long-term liver diseases”. These rules include many exceptions which are not common to enumerate or ambiguously created or difficult to fulfill in real life situations. Reasoning in realistic domains requires some kind of simplification or adaption to deal with exceptions or to increase the degree of belief in decision making situations [6].

Reasoning under uncertainty [7] is interchangeably associated with the handling of uncertain knowledge. The probability theory has an outstanding position and serves as basis for human's behaviour in decision situations where they reason under uncertainty. Next have a closer look at the nature of uncertain knowledge. In many application fields uncertainty is all-embracing involved. If a doctor tries to build a diagnosis (e.g. cardiac trouble) he might infer from a symptom (e.g. pain in the left arm). But not all patients with a specific symptom have cardiac problems (e.g. bronchia). In the case of specifying a causal (rule) dependence not all arm pains cause cardiac diseases. It is also possible that both disease states are unconnected. In practice, it is impossible to list the complete set of antecedents or consequents needed to ensure an exceptionless handling and the complete medical domain is too capacious to process all these sets.

A decision maker bypasses the enumeration exceptions by summarizing them in assigning each proposition to a numerical measure of uncertainty. Uncertainty solutions [8] must solve occurring questions in how to represent uncertain data, how to combine pieces of uncertain data and how to draw inference with uncertain data. Artificial Intelligence researchers tackle these problems in dealing with uncertainty using different techniques, mainly separated in quantitative and symbolic approaches. Quantitative approaches using certainty factors, fuzzy logic, belief functions, probabilities and possibilities [9]. Symbolic approaches dealing for instance with circumscriptions or default logic. In real-world decision scenarios, the best choice is providing a degree of belief based on the probability theory assigning numerical degrees in the interval [0,1]. Probability values correspond with a degree of belief that facts do or do not hold. The probability of a statement depends on the percepts a human has received to date. In uncertain reasoning, probability statements must indicate the evidence with respect to which the probability is being assigned [10]. If a human receives new or updates percepts, its probability assessments are updated to represent the new evidence. Based on probability theory a foundation for non-monotonic reasoning is given, whereas probabilistic networks allow the computation of quantified uncertainty. Statements in the form IF X then Y are quantifiable using probabilities.

### 3 Developed Solution for Integrating Uncertain Product Use Information

Figure 1 provides an overview of the developed overall solution for the feedback of product use information into product development. The upper half of the figure shows the situation during the product use phase for various customers. Every customer uses another instance of the product i within individual environmental conditions and load scenarios (1). Many customers maintain data bases with product use information for condition monitoring purposes, which include sensor data, environmental parameters, failures and incidences of maintenance, locally and isolated from each other (2).



**Fig. 1.** Overall solution for feeding back product use information into product development

So far, if not totally neglected, these product use information have been used for process optimization or for the prognosis of incidences of breakdown or failure. The approach described in this paper intends that the product use information is led back to the product development in the course of a feedback cycle. This serves as a basis for deducing room for improvement and optimization for new product generations.

For this, the data generated at various customer locations first has to flow back to the producer/supplier (lower half of the figure). The knowledge engineer edits the raw product use information with the help of knowledge discovery methods (3), deduces interrelationships between sensor data, environmental parameters, breakdowns and incidences of maintenance both qualitatively and quantitatively and finally manages the resulting diagnosis models (4) in the PLM data archive. In this context, the metadata model forming the core of an PLM system (5) has been extended to manage not only traditional product type but also product item data. (This aspect has been published in a further paper by the authors. It describes in detail the extension of a commercial PLM system for an integrated management of product item and product type data [17].)

By using knowledge-based methods within a PLM system it is possible to aggregate knowledge (6) from the data collected and edited with the help of knowledge discovery methods. The knowledge engineer can enter the product use information acquired from various customers in the PLM data model, generate individual diagnosis models and finally aggregate them. Thereby he can deduce a representative diagnosis model which takes differing environmental and load scenarios and their impacts on the machine condition, generally varying for individual customers, into consideration. The mentioned aspects of learning and aggregating individual diagnosis models based on machine learning and fusion algorithms were discussed on a technical level in [19] and will not be taken up again in detail within this paper.

On the basis of inference methods the product developer in collaboration with the knowledge engineer can interactively apply the aggregated diagnosis models in order to carry out simulations and „What-If“ analyses (7). For instance, definite load scenarios and environmental parameters can be set and depending on this probabilities for certain machine conditions, instances of breakdowns and recommended maintenance intervals can be calculated. All these simulations are based on data empirically gained during the product use phase. Such analyses form the basis for product developers to identify critical components and deduce room for improvement for new product generations. The topic of representing product use information and deducing knowledge on the basis of knowledge-based methods is the central theme of the following chapters.

The solution outlined in this section can be understood as an assistant for product developers. Deducing and realizing room for product improvement constitutes a creative process which cannot be automated with today's technologies. The aggregated diagnosis models along with analyzing techniques based on artificial intelligence methods should therefore support product developers in carrying out this creative process efficiently.

## 4 Representation of Product Use Knowledge

### 4.1 Requirements for the Knowledge Representation

The term product use information has already been distinguished from the purely subjective customer feedback in the introduction of this paper. Thus, no suggestions for improvement or positive/ negative reviews from customers as well as demands on future product generations expressed by users should be acquired and represented (These topics are already addressed in [21]). The focus of the represented data from the product use phase, which are to be considered within the scope of the present paper, is rather on objectively measurable information which accumulates during the use of a product.

These data strongly depend on the product examined. However, by concentrating on complex production machines and their components recurrent classes of data could be found. A structuring of the product use information to be represented into the following classes was conducted on the basis of the analysis of several case studies (stepper for the use of wafers in chip production, Wire Electrical Discharge Grinding (WEDG) machines for the electrical discharge machining of work pieces and other production machines).

Sensor data of the machine: the machine parameters captured in the course of a condition monitoring are part of this class. Examples are engine speed, consumption of operating materials, machine running times, voltages etc.

Environmental parameters: All objectively measurable ambient factors which have an influence on the operation of the examined machine fall into this category. Depending on the machine this can be, for instance, temperature, pressure, humidity of the ambient air.

Quality parameters of produced items: complementary to the already discussed sensor data of the examined machine monitored quality parameters of manufactured

items can (also via sensors) provide information about the condition of the machine. A concrete example is the proportion of functioning chips on a wafer.

Failures/ breakdowns: failures of components and other reduction of functioning are subsumed under this class.

Incidences of maintenance: maintenances and repair measures as well as the exchange of components are part of this group.

Knowledge representation and reasoning techniques are part of the field of artificial intelligence which is concerned with how knowledge can be represented symbolically and manipulated in an automated way through reasoning. The knowledge engineer dealing with the problem context at hand has to represent this knowledge in a formalized way based on an acquisition process for the acquisition and structuring of explicit and implicit knowledge from the product use phase. In which form can the gained sensor data, environmental parameters, breakdown data and incidences of maintenance be represented in order to use them in the processing stage for deducing coherences? Which demands have to be satisfied by the methodology for knowledge representation?

Hereafter, requirement potentials are discussed which are made on the representation and the management of product use information in order to be able to deduce knowledge-based improvement potentials from individual product entities and generally on the product type level.

Automatic transformation of PUI in PUK: In order to conclude improvement potentials from a knowledge-based feedback into the product development it is insufficient to provide product developers with the raw product use information (PUI) gained in the product use phase (e.g. condition monitoring data of a WEDG machine). Rather, these have to be edited intelligently and the product use knowledge (PUK) existing implicitly in the data has to be extracted. Therefore, one of the most important requirements on the methodology for knowledge representation is to enable deducing PUK as completely automated as possible from case data gained empirically during product use.

Integration of expert knowledge: The integration of a priori knowledge enables an integration of basic conditions or known dependencies in the preliminary stages of the model. For instance, by this, qualitative dependencies of load scenarios, environmental parameters or component breakdowns can be incorporated into the model as given expert knowledge.

Representation of uncertain knowledge: The representation of uncertain knowledge (as the result of incomplete, imprecise, inconsistent and defective data) and the handling of this type of knowledge is another important requirement especially with regard to the application domain ‘product use information’ since sensor data may be imprecise or missing and the interrelationships between sensor data, failures and incidences of maintenance contain a large degree of uncertainty, particularly in the case of new technologies.

Aggregation of PUK: Mechanisms of aggregation and fusion are necessary for the consolidation of individual models in order to ensure a higher representativeness and relevance of the resulting knowledge representation model.

Suitability for inference (‘What-If’ Analysis): The support of intelligent inference techniques are especially decisive with regard to the application of the methodology

for knowledge representation so that simulations and analyses (Which influence do certain environmental and load scenarios have on a certain component failure?) can be conducted on this basis.

Intuitive graphical visualization: Intuitive graphic means for the visualization of the derived product use knowledge are necessary in order to support product developers.

Model interpretability: The interpretability of the model and the conclusions assessed on the basis of inference techniques constitute a mental factor which should not be underestimated.

## 4.2 State of the Art of Methods and Techniques for Knowledge Representation

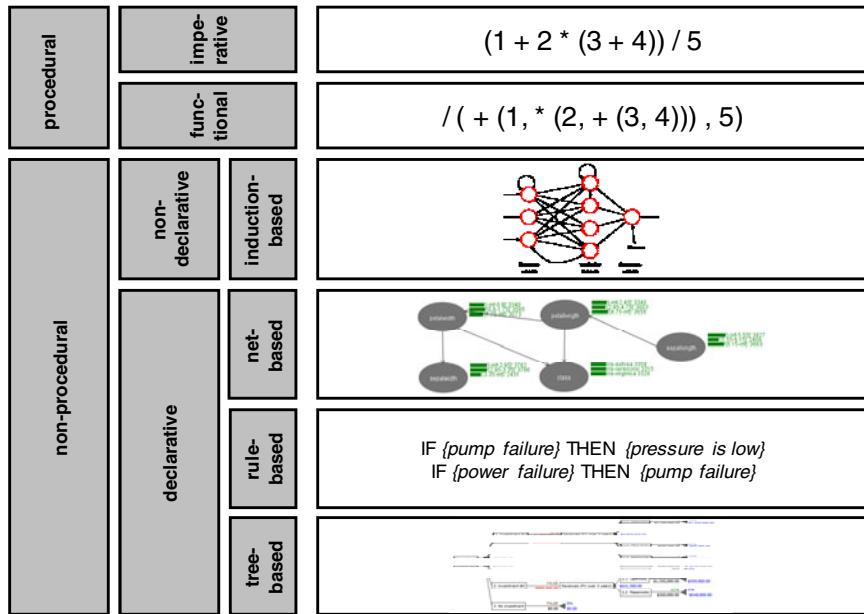
In the further course of this chapter models for knowledge representation are compactly presented and compared with each other. The models are structurally subdivided into various categories and the most important models of each category are presented. Afterwards, the most relevant approaches for the structured representation of product use information are evaluated. In this context, it is advisable to give a definition of knowledge representation before the individual knowledge representation models are introduced.

Knowledge representation is to be defined as a set of syntactic and semantic conventions for describing things and circumstances. The syntax specifies a set of rules which can be used for combining and grouping the symbols on which the knowledge handling is based. Thereby, expressions of the representation language can be formulated. The semantic describes the meaning of these expressions [22].

A way of structuring the most established knowledge representation forms is shown in Figure 2. In principle, it can be distinguished between procedural and non-procedural knowledge representation forms. Procedural knowledge representation forms are characterized by focusing on the description of procedures [23], while non-procedural knowledge representation forms rather concentrate on the individual knowledge elements and the relations between these elements [24].

A disadvantage of procedural knowledge representation forms often cited is the mixing of application-specific knowledge and general problem solution knowledge. Through this, the flexibility and the maintainability of such systems is strongly restricted, because a direct intervention in the program code becomes necessary with regard to an extension of the knowledge basis.

Non-procedural knowledge representation forms stand out due to a clear separation of general problem solution knowledge and application-specific knowledge. Rule-based systems constitute a good example. The domain-independent inference mechanisms (algorithms for the deduction of new facts through conclusions) are contained in the problem solution component while the actual domain knowledge is kept segregated in a rule data base. This separation allows the explicit mapping of knowledge in a program logic. However, the resulting disadvantage is a higher initial effort for creating such systems. On the other hand, it leads to a considerably improved maintainability. By depositing the case-specific expert knowledge within a closed knowledge basis it becomes exchangeable, upgradeable and modifiable without having to change the program code.



**Fig. 2.** Structuring of knowledge representation forms and examples

On the next granulation level non-procedural knowledge representation forms can be subdivided into declarative and non-declarative knowledge representation forms. An example for declarative knowledge representation forms has already been given above: rule-based systems.

Declarative knowledge representation forms like Bayesian networks are characterized by a presentation of facts and the relations among them in order to gain new knowledge on this basis [17]. With regard to the aspect of interpretability [18] conclusions can be transparently understood by the user. This aspect is not given for non-declarative approaches and should not be underestimated.

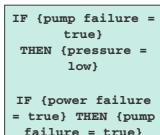
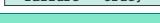
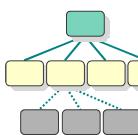
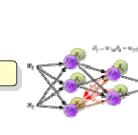
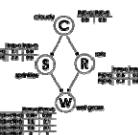
Artificial neural networks are a typical example for a non-declarative system. Artificial neural networks are capable of approximating functional coherences on the basis of case knowledge in form of training data sets [19]. Hereby, the interpretability by the user gets lost because it cannot explicitly be explained how neural networks arrive at certain conclusions [20]. This can also complicate the acceptance of systems based on such representation forms.

Procedural knowledge representation forms are inappropriate for modeling product use information and deduced knowledge because they concentrate rather on sequences than on the elements themselves and the relations between them. Therefore, the following evaluation focuses on rule-based systems (RBSs), Tree-Based Models (TBMs), Artificial Neural Networks (ANNs) and Bayesian Networks (BNs) as the most relevant non-procedural knowledge representation methods.

### 4.3 Evaluation of Methods for Knowledge Representation

The knowledge representation forms most relevant for modeling product use information with regard to the requirement criteria learning, integration of a priori knowledge, interpretability, inference, representation of uncertain and vague knowledge, visualization and aggregation techniques presented in 3.1 are compared and critically evaluated below (see table 1). Then, the methodology which fulfills the criteria for the knowledge-based processing of feedback information from the product use phase in the most target-oriented way will be applied in a practical scenario.

**Table 1.** Evaluation of various models for knowledge representation regarding the requirements made in chapter 4.1

	Rule-Based Systems	Tree-Based Models	Artificial Neural Networks	Bayesian Networks
Representation Requirements	 	 	 	 
Automatic PUK transformation	🟡	🔴	🟢	🟢
Integration of expert knowledge	🟢	🟢	🔴	🟢
Representation of uncertain knowledge	🟡	🟡	🟡	🟢
Aggregation of PUK	🟡	🔴	🟡	🟢
Suitability for inference („What-If“ Analysis)	🟢	🟡	🟢	🟢
Intuitive PUK graphical visualization	🟡	🟢	🔴	🟡
Model interpretability	🟢	🟢	🔴	🟢

Fulfillment of the requirements: ○ -not, ● -only to a small proportion, □ -partly, ■ -mostly, ■ -completely

With regard to automatic learning of knowledge on the basis of case data ANNs and BNs can play their advantages. Training data and appropriate learning algorithms (e.g. the backpropagation algorithm for ANNs [24]) enable learning general coherences from exemplary data sets as they occur in the field of product use information.

The integration of a priori expert knowledge turns out to be difficult in case of an ANN as the entire knowledge has to be learned on the basis of case studies. Serious disadvantages also arise in terms of options for the interpretation and visualization of knowledge. ANNs are not suitable for explaining an output semantically as the (artificial) neurons of the inner layers of an ANN in themselves do not possess a semantic interpretation and, hence, the ANN has to be regarded as a black box.

On the other hand, BNs offer the possibility to incorporate a priori expert knowledge in the model in addition to experimentally gained data. On a qualitative level dependencies between random variables can be modeled through manually integrated directed

edges while on a quantitative level conditional probabilities can be determined by experts on the basis of theoretical insights, empirical studies and subjective estimates.

Further advantages of Bayesian networks are the representation and processing possibilities of uncertain knowledge. In contrast to other examined knowledge representation forms it is not only possible to deduce the most probable diagnosis for a given set of symptoms, but also to determine the degree of uncertainty for the deduced conclusion. Furthermore, other possible diagnoses can be calculated according to descending probability. Inconsistencies, which for example occur for RBSs due to the local treatment of the factor uncertainty, can be avoided by a holistic examination.

RBSs, KNNs as well as BNs offer the possibility of inference and hereupon based “What-If” analyses. The requirements on visualization possibilities for product use information, however, are best fulfilled by BNs because of their clearly arranged representation of qualitative coherences in terms of a directed graph.

Especially relevant are possibilities for the aggregation of several knowledge representation models in order to be able to conclude general conclusions for the entire product class from the merged model. For the aggregation of Bayesian networks several concepts and algorithms are available [6]. Postulating that the same nodes in different networks also represent the same domains, an automated aggregation is made possible. Approaches for the individual weighting of several networks during the aggregation exist as well and contribute to the creation of a representative aggregated network. Concerning neural networks algorithms for aggregation exist as well (see e.g. [21]). Here, the same structure is assumed for all networks to be combined. The aggregation of several TBMs proves to be difficult unless there is not exactly the same topology for all trees and only associated probabilities are to be aggregated. The aggregation of several rule bases turns out to be complex as it has to be tested in a holistic approach whether additional rules lead to inconsistencies. In case of similar rules it then has to be individually determined which rules should be incorporated into the aggregated rule basis.

As measured by the requirements put forward in chapter 3.1 Bayesian networks overall appear as the most promising knowledge representation form for modeling product use information and deduced knowledge especially with regard to aggregation, interpretation, inference and visualization possibilities.

## 5 Proposed Knowledge-Based Approach

### 5.1 Use Case Scenario

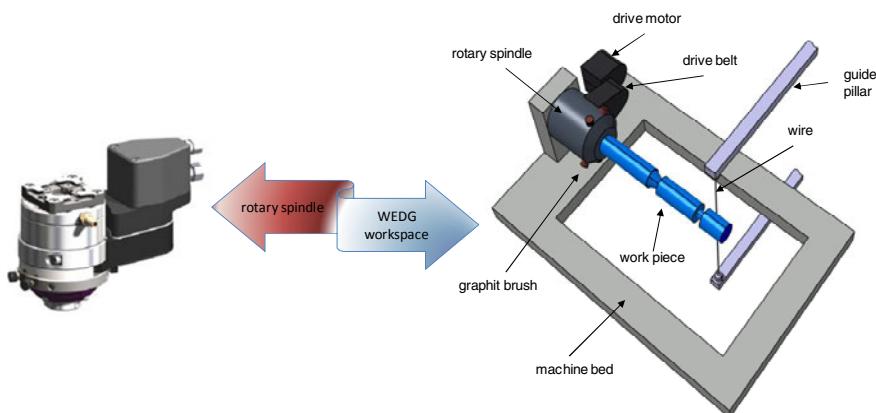
In engineering a vast number of manufacturing methods exists which are applied by several machine tools to handle a work piece. In general, parts are formed by archetyping, transforming, disconnecting, assembling, coating or changing certain material properties.

The procedure of erosion belongs to the disconnecting procedures since, during the process of manufacturing, the cohesion of the work piece gets changed. Within the disconnecting procedures a further breakdown is carried out so that a certain type of cohesion modification is reflected. Thereby the spark erosion is assigned to the erosive procedures. Erosive procedures are characterized by the way they do not perform

any mechanical action during the adaptation of the work piece. This enables a handling which takes place completely independently of any attributes of the work piece, for example without consideration of a work piece's hardness.

The principle of spark erosion is based, as the name suggests, on sparks and their thermal consequences on a work piece. Particles are separated by the sparks and afterwards removed by mechanical and/or electro-magnetic power. This process is also called Electrical Discharge Machining (EDM) [22].

Thereby the sparks emerge via electrical discharges between certain tools and the work piece and, furthermore, create a high temperature at the point of working. Besides the material removal on a work piece, additionally, there is a noticeable metal removal on the working tool. For cooling purposes and for the removal of segregated material there is a dielectric fluid. It is characterized by an especially poor conductivity and so isolates the electric wire and the work piece.



**Fig. 3.** Rotary spindle and schematic representation of the WEDG workspace

Next, the general structure of a wire-electro discharge machine is presented. In these machines an implementing electrode in form of a wire is used, thus contact-free consigning its image on a work piece. Since the wire itself thereby incurs a certain material removal, it is continuously replenished by an engine in order to provide a constant material removal on the work piece. A generator supplies the working tool and the work piece with required voltage, so that a discharge and the associated appearance of erosive sparks becomes at all possible at the working point. The relative movement of wire guide and work piece is taken on by a separate control.

EDM machines working with rotary spindles are commonly called Wire Electrical Discharge Grinding (WEDG). These machines are mostly used in the manufacturing of micro-structured work pieces. Figure 3 illustrates the structure of a rotary spindle. The rotary spindle is assembled on the machine bed and put into motion by a drive motor with a drive belt. The electrode (work piece) is fixed to the spindle with some kind of chuck and therefore rotates with exactly the same angular speed. The speed is specified by the drive motor, which, by its own, is regulated by a control instance. Simultaneously, the erosion wire is continually run by a wire guide. This represents the antipole for the electrode. To enable the accruement of erosion sparks the working

point is continually supplied with dielectrics. The electrode on the other side gets its power from graphite brushes which force the current conduction inside the ball bearing of the rotary spindle to decrease. The drive motor's electric circuit is strictly disconnected from the electric circuit used for the erosion by an insulating plate. The rotary spindle and drive motor are continually provided with compressed air, so that any intrusion of liquids or removed material is avoided.

## 5.2 Bayesian Networks for Modeling Product Use Knowledge

Bayesian networks can model dependencies between incidences like e.g. breakdowns or maintenance adequately on the basis of probabilistic constructs. Here, a Bayesian network represents a causal or probabilistic net which is appropriate for representing uncertain knowledge and resulting possible conclusions. It consists of a directed acyclic graph (DAG) in which nodes represent incidences as random variables and directed edges represent conditional dependencies. Every node is given a conditional probability distribution of the random variable it represents. If new critical values appear, updated probability distributions of other random variables can be calculated by means of dedicated nodes in the Bayesian network.

A Bayesian network consists of a qualitative structural and quantitative numeric component. The qualitative component represents the coherences between the random variables of the problem scenario as well as the dependencies between product use information (like conditional dependent, independent) expressed through the graph-based structure. A Bayesian network can compactly describe the common probability distribution of all involved random variables by using known conditional independencies. Qualitatively, relations of dependencies and independencies can be depicted. Every random variable  $X_i$ , which possesses finitely many conditions  $x_1, x_2, \dots, x_n$ , is allocated a table of conditional probability distributions for every possible combination of conditions  $a, \dots, z$  of the parent nodes  $A, \dots, Z$  of  $X_i$  as follows:

$$P(X_i = x_i | a, \dots, z) \quad \forall i = 1, \dots, n. \quad (1)$$

Regarding especially the root nodes one is concerned with unconditional probability distributions as a priori distributions. If a Bayesian network consists of  $n$  random variables  $X_1, X_2, \dots, X_n$ , the common probability distribution of all nodes can be expressed as follows:

$$P(X_1, X_2, \dots, X_n) = \prod_i P(X_i | pa(X_i)). \quad (2)$$

The direct parent nodes of the random variables  $X_i$  will be expressed in this context as  $pa(X_i)$ . The example of the abovementioned rotary spindle presented in Figure 4 serves as illustration. Such a network structure can be achieved on the basis of empirically gained data with the help of a qualitative algorithm for learning structures of Bayesian networks [6]. For motivation and clearly arranged graphic visualization the rotary spindle, first consisting of the three random variables

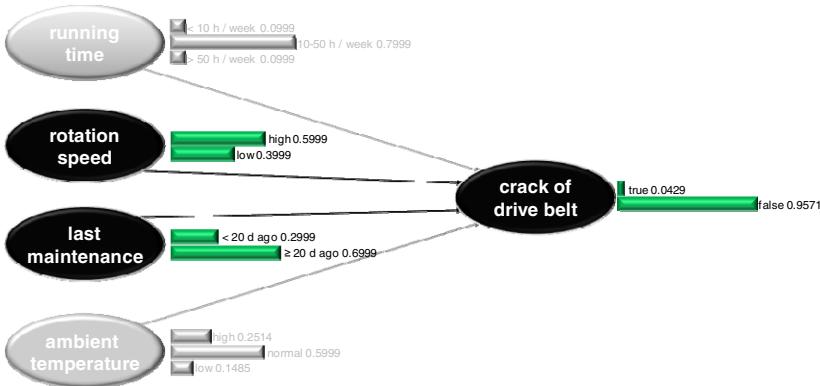
- R: rotation speed
- M: last maintenance
- B: crack of drive belt,

will be indicated with 2 conditions for each random variable.

The node crack of drive belt (B) has the two parent nodes rotation speed (R) and last maintenance (M). In this rotary spindle net the node crack of drive belt has two conditions: true (t) and false (f). Node rotation speed has the two conditions high (h) and low (l) and node last maintenance has the two conditions less20d (l20) und greaterequal20d (ge20). The qualitative component of the Bayesian network is already given in Figure 4.

The quantitative component in this scenario consists of the a priori probabilities  $P(R=h)$ ,  $P(R=l)$ ,  $P(M=l20)$  and  $P(M=ge20)$  in the root nodes as well as the conditional probability tables (CPTs) which are exemplarily represented for the node crack of drive belt as follows:

$$\begin{aligned} & P(B = t | R = h, M = l20) \quad P(B = t | R = h, M = ge20) \quad P(B = t | R = l, M = l20) \quad P(B = t | R = l, M = ge20) \\ & P(B = f | R = h, M = l20) \quad P(B = f | R = h, M = ge20) \quad P(B = f | R = l, M = l20) \quad P(B = f | R = l, M = ge20) \end{aligned} \quad (3)$$



**Fig. 4.** Influencing random variables causing the failure “crack of drive belt”

Belief values represent the confidence that a given node is in a certain condition. The initial belief value  $\text{Bel}(R)$  is given for the root node R for example through the a priori probability  $P(R)$ . For arbitrary nodes X the belief value  $\text{Bel}(X)$  can be declared as  $P(X|O_X)$  in which  $O_X$  describes all nodes except X. A Bayesian network is initialized as soon as all belief values are calculated. If new knowledge is available, the belief values have to be updated for all nodes. Efficient algorithms exist for the propagation of knowledge (compare [22]).

To calculate the overall probability that node crack of drive belt is in state true the CPT of the node crack of drive belt and also the two parent nodes, rotation speed (high/low) and last maintenance (l20/ge20), are required. Thereby  $P(B=t)$  can be calculated as follows:

$$\begin{aligned} P(B = t) &= P(B = t | R = h, M = l20) \cdot P(R = h) \cdot P(M = l20) \\ &\quad + P(B = t | R = h, M = ge20) \cdot P(R = h) \cdot P(M = ge20) \\ &\quad + P(B = t | R = l, M = l20) \cdot P(R = l) \cdot P(M = l20) \\ &\quad + P(B = t | R = l, M = ge20) \cdot P(R = l) \cdot P(M = ge20). \end{aligned} \quad (4)$$

The introduced spindle scenario can be extended to 5 random variables by adding further product use information like the environmental parameter temperature and the

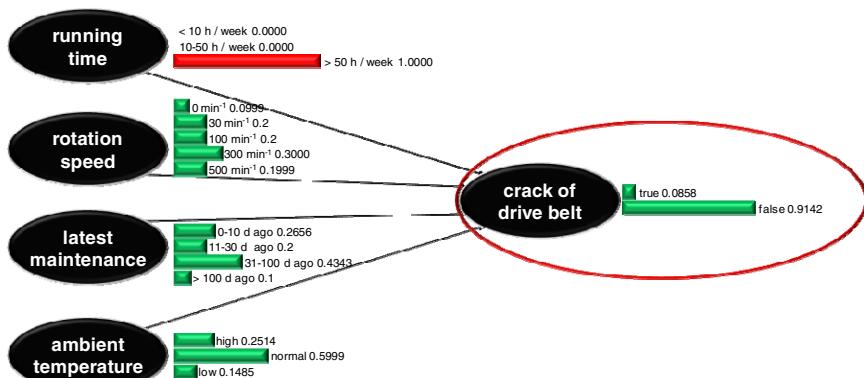
sensor parameter machine running time. Through information granulation the characteristics of the conditions of the random variables can be fine or coarse granularly be realized (e.g. extension of the conditions of the random variable rotation speed from two (low/ high) to five (0 min<sup>-1</sup>, 30 min<sup>-1</sup>, 100 min<sup>-1</sup> from low and 300 min<sup>-1</sup>, 500 min<sup>-1</sup> from high) values). Here, it is essential to determine a balanced standard between inference with regard to interpretation time and the complexity concerning the CPT assignments of the represented Bayesian network. The rotary spindle network extended to 5 random variables cannot only be used for deducing and comparing quantity measures (for instance for determining which breakdowns happen most frequently) but also for investigating more closely coherences between identified critical components and load, maintenance and environmental scenarios on the basis of “What-If” analyses.

### 5.3 “What-If”-Analysis to Deduce Room for Product Improvements

Besides averaged distributions on how susceptible individual components are and under which load scenarios and environmental parameters the spindle operates at various customer locations, the model can also be used for simulation purposes. After identifying the failures or breakdowns which occur most frequently it is possible to conduct a detailed analysis on which factors influence a certain breakdown.

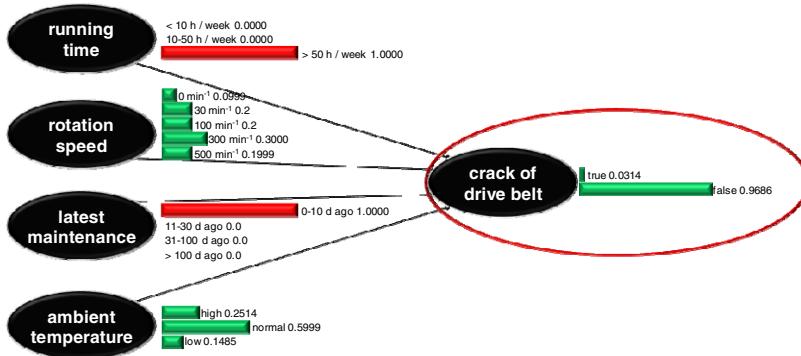
Coherences between sensor data of the rotary spindle (rotation speed), incidences of maintenance over time (last maintenance) and breakdowns of individual rotary spindle components (crack of drive belt) can be deduced on the qualitative as well as the quantitative level. On which parameters do the different breakdowns depend qualitatively? How does the probability for the cracked drive belt quantitatively change in the extended example from chapter 4.1, if the spindle is used more than 50 hours a week?

These and similar questions can be answered on the basis of an underlying inference engine by applying evidences like “running time of the spindle more than 50 hours a week” (compare red bar in Figure 5) and propagating them in the network. The outcome of this is a doubling of error probability with regard to the crack of the drive belt on the basis of acquired product use information.



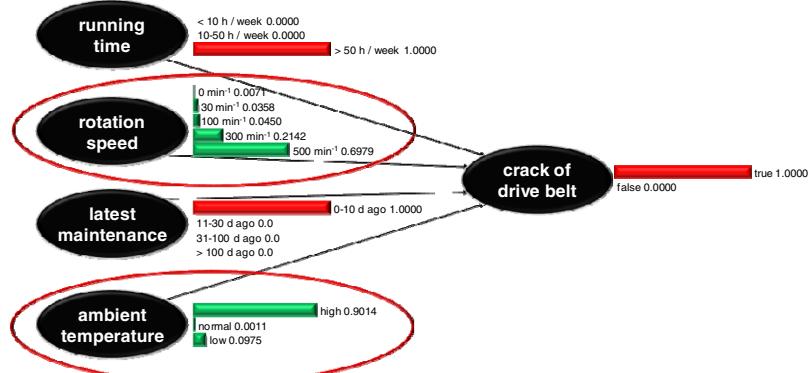
**Fig. 5.** Influence of the weekly running time on a crack of the drive belt

However, if it is additionally known that the last maintenance has taken place 0-10 days ago, the probability for a crack of the drive belt decreases to 3,14% (see Figure 6).



**Fig. 6.** Influence of a running time of more than 50 hours a week on a crack of the drive belt in case the last maintenance has taken place 0-10 days ago

In case the drive belt cracks nevertheless, on the basis of the acquired data material one can assume a high probability that the rotation speed and/ or the ambient temperature have been high (compare Figure 7).



**Fig. 7.** Most probable causes for a crack of the drive belt with regard to the set assignments for running time and date of last maintenance

While the given example based upon fictitious data sets only serves the purpose of clarifying the potential of the introduced framework by means of an easily understandable example, in practice a significantly wider range of product use information accumulates. In contrast to the given example, many of the relations between machine sensor data, environmental parameters, breakdowns and incidences of maintenance can neither be known on a qualitative nor on a quantitative level.

On the basis of machine learning algorithms and appropriate aggregation methods as presented in [6] coherences between product use information can be revealed and represented as a Bayesian network. With the help of the presented approach scenarios can also be simulated in case of complex networks in the course of a „What-If“ analysis and their impacts on relevant variables can be examined in order to support the product developer in deducing improvement potentials [25,26].

To sum up, knowledge representation models automatically established on the basis of empirically acquired product use information (below also diagnosis models) offer the following possibilities:

- Identification of the components breaking down most frequently
- Clearly arranged graphic visualization of qualitative coherences
- Discovering factors (scenarios of ambience, load and maintenance) that have an influence on certain breakdowns
- Deduction of quantitative dependencies on the basis of empiric product use information

Especially in case of complex machines being based on technologies which yet cannot be entirely “understood” the processed diagnosis models can serve as a basis for the deduction of design, implementation and use guidelines.

## 6 Conclusions and Outlook

The concept for a knowledge-based feedback of product use information into product development, as described in the first part of the paper, has been prototypically implemented. In this context, the PLM solution Teamcenter Engineering of Siemens PLM Software was chosen as a testbed. The functionality was enhanced with regard to creating and modifying product items, their association to the appropriate product type, linking maintenance events, condition monitoring data and diagnosis models, including support for visualization and “What-If” analyses [11].

Human tasks require intelligent behavior with some degree of uncertainty. A knowledge-based solution exhibits such intelligent behavior by modeling the empirical associations and heuristic relationships that (mechanical domain) experts have built over time. Types of uncertainty that can occur may be caused by different problems with the data, which might be missing, imprecise, inconsistent or unavailable. Uncertainty may also be caused by the represented knowledge since it might not be appropriate in all situations.

For the appropriate representation of domain knowledge, which should also comprise breakdowns, incidences of maintenance and the dependencies between all involved elements besides sensor data and environmental parameters, in the second part of the paper various knowledge representation forms have been considered and critically evaluated with regard to the proposed requirements. Bayesian networks have proven to be the most promising model, especially in view of aggregation, interpretation, inference and visualization possibilities.

In order to evaluate the fundamental practicability of the concept an exemplary scenario has been chosen, which describes the essential steps for the knowledge-based feedback of product use information.

As the success of the feedback concept strongly depends on the willingness of the customers to provide individual product use information, additionally, there is a high demand for a motivation concept for customers today. Nevertheless the emerging trend at manufacturing companies as well as service providers to break with the traditional product and service understanding and to address the integrated consideration of products and services as customer-oriented overall solutions (Industrial Product-Service Systems (IPS<sup>2</sup>) [24]) will enable IPS<sup>2</sup> providers to get easy access to information generated in the product use phase at various customer locations. In effect this will facilitate the industrial implementation of the presented solution approach.

The paper at hand focuses explicitly the product area and excludes the service sector from consideration. However, the observable trend towards IPS<sup>2</sup> will demand an integration of additional types of feedback in the PLM concept. Basically, three different types of feedback can be acquired in addition to an IPS<sup>2</sup>: First, product-related feedback, second, service-related feedback and third, IPS<sup>2</sup>-related feedback.

Within the realms of product-related feedback active feedback (subjective requirements, reviews, customer satisfactions) [21] and passive feedback (objectively measurable product use information) dealt within the present paper can be distinguished. Principally, this structure can also be transferred to service-related feedback. However, it is insufficient to cover feedback which can neither be directly allocated to products nor services.

Such an IPS<sup>2</sup>-related feedback could be, for instance, the request for a greater availability of a machine in the course of an availability-oriented business model [3], which adequately induces certain product and service adaptations. The consideration of such mutual relations between product and service shares at the moment is still an object of basic research. In this context the concept for the knowledge-based feedback of product use information into product development presented in the paper at hand is one of the puzzle pieces necessary for an IPS<sup>2</sup> Feedback Management.

## References

1. ElMaraghy, H., ElMaraghy, W.: Advances in Design. Advanced Series in Manufacturing. Springer, London (2006)
2. Kota, S., Chakrabarti, A.: Development of a Method for Estimating Uncertainty in Evaluation of environmental Impacts during Design. In: 16th International Conference on Engineering Design (ICED 2007), Paris, France (August 2007)
3. Meier, H., Kortmann, D.: Leadership - From Technology to Use; Operation Fields and Solution Approaches for the Automation of Service Processes of Industrial Product-Service-Systems. In: 14th CIRP Conference on Life Cycle Engineering, LCE 2007, Tokyo, Japan, pp. 159–163 (June 2007)
4. Abramovici, M.: Future Trends in Product Lifecycle Management. In: 17th CIRP Design Conference, Berlin, Germany (March 2007)
5. Viertl, R., Hareter, D.: Beschreibung und Analyse unscharfer Information, 1st edn. Springer, Wien (2006)
6. Neapolitan, R.E.: Probabilistic Reasoning in Expert Systems. J. Wiley & Sons, New York (1990)
7. Brachman, R.J., Levesque, H.J.: Knowledge Representation and Reasoning. Morgan Kaufmann, San Francisco (2004)

8. Luini, L.: *Uncertain Decisions. Bridging Theory and Experiments*, 2nd edn. Kluwer Academic Publishers, Dordrecht (2007)
9. Voorbraak, F.: Reasoning with Uncertainty in Artificial Intelligence. In: Dorst, L., Voorbraak, F., van Lambalgen, M. (eds.) RUR 1995. LNCS, vol. 1093, pp. 52–90. Springer, Heidelberg (1996)
10. Fisseler, J.: *Learning and Modeling with Probabilistic Conditional Logic*. IOS Press, Amsterdam (2010)
11. Abramovici, M., Neubach, M., Fathi, M., Holland, A.: Enhancing a PLM System in Regard to the Integrated Management of Product Item and Product Type Data. In: IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2008), Singapore (October 2008)
12. Abramovici, M., Neubach, M., Fathi, M., Holland, A.: Competing Fusion for Bayesian Applications. In: 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2008), Málaga, Spain, pp. 378–385 (June 2008)
13. Abramovici, M., Schulte, S.: Optimising Customer Satisfaction by Integrating the Customer's Voice into Product Development. In: 16th International Conference on Engineering Design (ICED 2007), Paris, France (August 2007)
14. Spur, G., Krause, F.-L.: *Das virtuelle Produkt – Management der CAD-Technik*. Carl Hanser Verlag, Munich (1997)
15. Bullinger, H.-J., Warschat, J., Lay, K.: *Künstliche Intelligenz in Konstruktion und Arbeitsplanung*. Verlag Moderne Industrie, Landsberg/Lech (1989)
16. Russel, S., Norvig, P.: *Artificial Intelligence. A Modern Approach*. Prentice Hall, New Jersey (2003)
17. Mertens, P.: *Expertensysteme in der Produktion*. Oldenbourg, Munich (1990)
18. Goertzel, B.: Probabilistic Logic Networks. In: *A Comprehensive Framework for Uncertain Inference*. Springer, New York (2008)
19. Haykin, S.: *Neural Networks and Learning Machines*. Prentice-Hall, New Jersey (2008)
20. Zheng, W., Zhang, J.: Advances in Neural Network Research and Applications. In: Zhang, L., Lu, B.-L., Kwok, J. (eds.) ISNN 2010. LNCS, vol. 6063, pp. 684–692. Springer, Heidelberg (2010)
21. Granitto, P.M., Verdes, P.F., Ceccatto, H.A.: Neural Network Ensembles: Evaluation of Aggregation Algorithms. *Artificial Intelligence* 163(2), 139–162 (2005)
22. Weck, M., Brecher, C.: *Werkzeugmaschinen - Maschinenarten und Anwendungsbereiche*. Springer, Heidelberg (2005)
23. Spiegelhalter, D.J., Lauritzen, S.L.: *Probabilistic Networks and Expert Systems*. Springer, Heidelberg (1999)
24. Meier, H., Uhlmann, E., Kortmann, D.: Hybride Leistungsbündel – Nutzenorientiertes Produktverständnis durch interferierende Sach- und Dienstleistungen. In: *wt Werkstattstechnik online*, 7/2005. Springer-VDI-Verlag, Düsseldorf (2005)
25. Neapolitan, R.E.: *Learning Bayesian Networks*. Prentice-Hall, New Jersey (2004)
26. Darwiche, A.: *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, UK (2009)

# Authentication Using Multi-level Social Networks

Justin Zhan and Xing Fang

National Center for the Protection of Financial Infrastructure  
Madison, South Dakota, U.S.A.  
`{Justinzzhan,xingfang912}@gmail.com`

**Abstract.** Authentication is an important way to protect security. Many web applications such as mobile banking, emails, online shopping require users to provide their credentials to authenticate themselves before they can use the services. Four factors, including password, token, biometrics, social networks, have been proposed for authentication. However, the proposed authentication schemes of the four factors all suffer from different shortcomings. In this paper, we propose a multi-level social authentication framework. Our analysis shows that the framework is much more robust than the existing approaches. More importantly, we minimize the potential privacy disclosure of users during the authentication procedure. We present our framework, conduct performance analysis and various security attack analysis.

**Keywords:** Multi-level, Authentication, Human factors.

## 1 Introduction

Web applications like mobile banking, shopping, and email systems generally require users to provide credentials to authenticate themselves. Usually, the credentials include a username and a password. To have a better security, a token code such as RSA's SecureID, is needed in addition to the password. Moreover, biometrics such as fingerprints, iris scans, and voice recognition is another factor for authentication. Hence, password, token code, and biometrics have been considered as three primary authentication factors that are widely adopted by traditional authentication systems [1].

In general, the more secure the system is, the more complex the system is. This makes two-factor authentication method a popular solution for many current authentication systems in which a password together with either a token code or biometrics are required for authentication. This two-factor scheme works well but it makes users suffer from managing their hardware token devices due to frequently lose, break, or forget them. Biometrics also presents its weakness in that broken fingers or loss of voice are able to temporarily hinder the authentication.

A fourth factor authentication method via social network is introduced by RSA Lab [1] as an emergent way to authenticate a user when he cannot get access to the fresh token. In this vouching approach, the user needs to contact with his friend for help. The friend, also identified as a helper, should recognize the user and then issue him a vouch code.

Soleymani and Maheswaran [2] suggest using mobile phones to automate the vouching process. Instead of the vouch code, a vouch token is automatically generated by a

helper's mobile phone and issued to a user's mobile phone after a call or a Bluetooth sighting. Once enough tokens have been collected, the user is able to send them to the central server to authenticate himself.

In this paper, we introduce a novel framework for social authentication, where the bulk of the social authentication work is imposed on an authentication server through a multi-level helper-list. Our contributions are as follows: (1) It significantly automates social authentication process by maintaining a multi-level helper list on the authentication server. When the server receives an authentication requirement from a user, it will randomly select a certain number of helpers from the list and contact them via a mobile phone network. (2) It reduces user-side burdens. Users only need to communicate with the authentication server through their mobile phones rather than making multiple phone calls or maintaining several tokens. (3) It assures the completion of social authentication process by leaving an ultimate chance which allows users to be self-authenticated.

In the following section, we review previous research work in social authentication. In section 3, we introduce our framework and the relevant authentication algorithms. We also provide a thorough analysis on the performance of the framework in section 4 and expand the analysis in section 5 to enhance the efficiency. Section 6 discusses security and attack analysis where the security of our framework is evaluated based on analyzing multiple attacks.

## 2 Related Work

### A. Human Factors and Social Authentication Protocols

Human factors are inevitably involved in secure authentication protocols. Under a password authentication protocol, a user needs to remember and maintain an appropriate password in order to login a system. In this scenario, human meditation is a factor that dominates the success of the authentication process. Prior research shows that this factor is somewhat unreliable since people occasionally forget their passwords or let them easily to be acknowledged by the acquaintances [3, 4]. However, some researchers did find out that fundamental human interactions can be useful in developing security protocols. Ellison [5] has pointed out the importance of human behavior in interacting with cryptographic authentication protocols. McCune et al. [6] have introduced "Seeing is Believing", an authentication protocol that requires human visual verification on two dimensional code bars which contain cryptographic key materials. Brainard et al. [1] provided a very first social authentication protocol that is developed based upon the study of human factors. Even though it is the first time for such a protocol to emerge, its theory is by no means new to us. For instance, when you are introducing one of your friends to your parents, a social authentication is being conducted. Similar with the approach in [1], a social authentication system, which is used as a backup mechanism to regain access to accounts, is introduced by Schechter et al. [7]. Users of this system are asked to contact with a certain number of pre-assigned trustees to get account-recovery codes. Both helpers in [1] and trustees in [7] are considered as human factors. The relationships between users and those trustees (helpers) consist of a collection of networks, namely social networks, upon which social authentications protocols are deployed. A recent work [8] reveals a photo-based social authentication

approach via social networks, where a photo is presented to a user, in which the photo includes a group of people uniquely known by the user. To be authenticated, the user is required to identify names of subjects on the photo. The vouch token, account recovery codes, and names of subjects are proofs for social relationships. Eligible users of social authentication protocols should be capable to possess them. For the authentication protocols in [1, 7], proof is issued by human. A security concern that how to validate a user's identity before issuing the user proofs is then presented. In [1], a helper should be able to recognize the user's voice or appearance. In [2], a user has to make the duration of a phone call long enough in order to receive a vouch token, whereas in [8], proofs are only held by the eligible users since it is highly unlikely for an ineligible user knowing all of the subjects' names.

#### *B. Mobile Devices as an Authenticator*

People always carry mobile devices such as mobile phones in their daily life. Nowadays, mobile phones are the vehicles to convey instant communications. This characteristic makes a mobile phone an ideal authenticator. One of the applications in making use of mobile phones for authentication is to send One Time Password (OTP) through the Short Message Service (SMS). However, the unencrypted SMS message is subject to be revealed to eavesdroppers. A recent research work [9] explores a new method that allows mobile phone becoming a stand-alone hardware token, such as a SecureID, by running dynamic password generation software on it. Along with the rapid development on mobile phone network and technologies, Public-key Infrastructure (PKI) has been able to be implemented on mobile phones. By integrating a Trust Platform Module (TPM) [10], both symmetric and asymmetric cryptography algorithms can be deployed on the mobile phone. The possibility of PKI's usage on mobile phones provides a more secure solution for authentication.

The vouching attribute of social authentication brings us a question: what happens if all of the helpers are unavailable? There is no guarantee on a successful authentication in [1], if all of the helpers are not able to answer the phone calls or receive the Bluetooth sightings. Similarly, the authentication process in [2] will not be accomplished if all of the helpers cannot talk long enough with the user. Our framework solves the problem by implementing the multi-level helper list. It allows users to conduct self-authentication as an ultimately solution in the case of all the helpers are unavailable to authenticate the users.

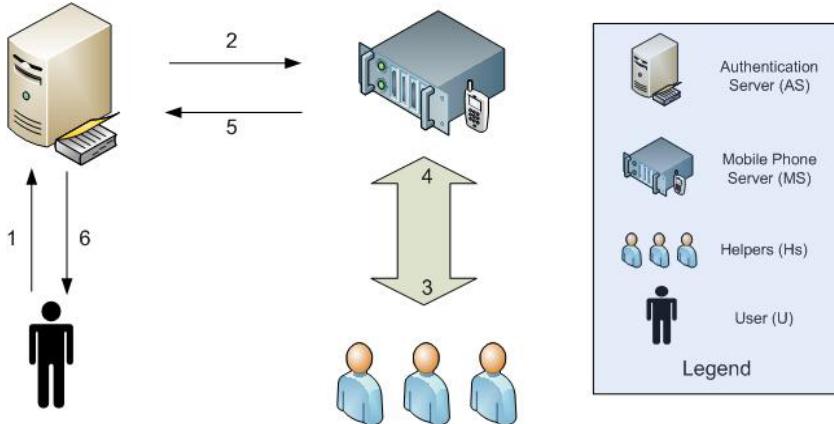
## **3 The Framework**

#### *C. Framework Layout*

Our social authentication framework is appropriate to be implemented under the following conditions:

- In the two-factor authentication scenario where a password and a token code are required.
- A user is able to remember his password but lost his token device or unable to access the fresh token code.
- This user has a multi-level helper list pre-stored on an authentication server.

- The authentication server has a webpage designed to retrieve username and password in order to allow the user to start a social authentication process.
- This authentication server stores helpers' public keys as well as its public-private key pair.



**Fig. 1.** Framework Layout

The above figure provides a layout of our framework. The framework has four components: User (U), Authentication Server (AS), Mobile Phone Server (MS), and Helpers (Hs). This framework's entire operational procedures are described as follows:

1. U initiates a social authentication process by submitting his user ID and password to AS.
2. AS chooses a group of Hs from the pre-stored list and sends messages to MS.
3. MS forwards the messages to Hs.
4. Hs give feedbacks to MS.
5. MS forwards the feedbacks to AS.
6. AS calculates if U can be authenticated, based on the feedbacks, then responds to U.

#### D. Helper Selection and An Example of A Multi-level Helper List

The essence of social authentication is trustworthy relationships from which helpers or trustees are chosen. Friendship is a subjective social relationship that can be compromised at any time [8]. However, it is one of the major social relationships. On the other hand, we claim that a kinship commonly holds more affinity than a friendship does, because of the blood tie. Thus, we suggest that our example of a multi-level helper list is layered based on the affinity between a user and helpers. The closer a relationship is the higher level a helper is.

Table 1 is the example list. The lowest level is #4 and the highest level is #1. Each of the levels excluding level #1 stores information of helpers including their names and mobile phone numbers. Level #1 stores a user's information. Besides, a certain number of life questions of the user are stored on the highest level. Consequently, this provides an option which makes a user conduct self-authentication by answering all of the life questions correctly.

**Table 1.** An example of a Multi-level helper list

Levels	Number of Helper Selected
Level #1 User's Name, Mobile Phone Number and Life Questions	1
Level #2 User's Relatives Names and Mobile Phone Numbers	2
Level #3 User's Buddies' Names and Mobile Phone Numbers	3
Level #4 User's General Friends' Names and Mobile Phone Numbers	4

We notice that a successful authentication is achieved only if enough required credentials have been collected. Hence, each of the levels must be able to, at least, provide equal amount of credentials required for a successful authentication, because any level can be assigned for deploying authentications. To organize how many helpers that should be selected at a specific level, we take the following two conditions into account: (1) The magnitude of credentials that one helper has is directly related to the helper's social relationship with the user. (2) From the perspective of social authentication, it is intuitive that the more affinity with the user, the more magnitude of credentials the helper holds.

Based on the layout of our example list, Level #1 is designed for the user to conduct self-authentication. The total amounts of credentials are solely provided by the user. Correspondingly, we allow two, three, four helpers to be selected at Level #2, #3, and #4, respectively. Therefore, one helper at Level #2 holds one half amounts of credentials; one helper at Level #3 holds one third amounts of credentials; one helper at Level #4 holds one quarter amounts of credentials.

#### E. Authentication Protocol Steps

Before going through the details of each authentication protocol step, there are some notations needed to be clarified. Note that we assume all of the helpers' public keys are stored on the central authentication server together with their digital certificate [12].

$T$	A readable message
$R_S$	Random numbers generated by the authentication server
$PV_S$	Authentication server's private key
$PK_S$	Authentication server's public key
$PV_H$	Helper's private key
$PK_H$	Helper's public key
$PK_U$	User's public key
$H()$	Secure hash function
$\{data\}Key$	Asymmetric encryption of $data$ using $Key$

The authentication protocol proceeds as follows:

1. A user opens a specific web page named “Social Authentication Page” within https tunnel. The web page prompts the user to type in his username and password.

2. This user inputs the required information and clicks the submit button to initiate an authentication process.
3. The username and password collected on step 2 are sent to the authentication server to first authenticate the user's identity.
4. Based upon a successful authentication of the user's identity, the server picks up a certain number of helpers from the default level of the list and sends a message to each of them simultaneously. The pattern of each message is  $\{\{T, R_S\}PV_S\}PK_H$ .
5. After a helper receives the message, it can be decrypted by the helper's private key then the server's public key. The decryption requires the helper input her mobile phone PIN.
6. Based upon the successful decryption on step 5, the helper reads the message  $T$  and confirms the authentication to the user. A feedback message under the following pattern is sent to the server  $\{\{H(T, R_S)\}PV_H\}PK_S$ .
7. Once the server collects enough feedbacks from helpers, a temporary token code is generated and sent to the user within the following message pattern  $\{\{code\}PV_S\}PK_U$ .
8. User is then prompted to input his mobile phone PIN to decrypt the message and retrieve the temporary token code.
9. User uses his password along with the temporary token code to login under the regular two-factor authentication web page.

#### *F. Authentication Algorithm and Analysis*

Given the example list, we assume that helpers are selected from the lowest level. If a helper is able to give a feedback within the period of time, a positive flag will show up on the server. Once there are enough positive flags, the user is eligible to be authenticated. However, a negative flag shows up because the helper did not response in time. It implicates that either she did not read and response the message timely or refused to response it. In this case, the following is the algorithm to execute the authentication:

- Server has all positive flags. The user is ready to be authenticated.
- Server has all negative flags. In this situation, server will choose the other equivalent number of helpers from the same level.
- Server has N positive flags, where  $0 < N <$  Required number of helpers. On Level 4, if N equals 3, the user is eligible to be authenticated due to the majority vote; if N equals 2, the server will select the other two people from the same level and send them the same message; if N equals 1, the other three people will be selected. On Level 3, if N equals 2, the user is authenticated because of the majority vote; if N equals 1, the other two helpers are selected by the server. On Level 2 there is no majority vote, therefore if N equals 1, another helper will be chosen.
- Server is allowed for 3 tries in selecting helpers on the Level#4, #3, and #2. Accordingly, there is always possible that there are not enough positive flags to authenticate the user after 3 tries. At this point, the server will turn to select helpers on higher levels. Authentication is then transformed from collecting single level feedbacks to calculating multiple level feedbacks, because of the magnitude of credentials contained in one feedback is varied based on different levels. For instance, suppose there are two feedbacks collected from Level 4 and one feedback from Level 3. The total number of feedbacks should not be the direct sum of feedbacks

from both levels. Hence, to precisely calculate feedbacks based on multiple levels, a group of authentication functions are introduced: We use  $f(n)L_N$  to denote an authentication function at Level N. The argument 'n' represents the number of feedbacks at Level N. These functions are demonstrated as:

- ◆  $f(n)L_4 = n$
- ◆  $f(n)L_3 = n + 3/4 * f(n)L_4$
- ◆  $f(n)L_2 = n + 2/3 * f(n)L_3$

Each of the functions has an authentication requirement:  $f(n)L_4 \geq 3$ ;  $f(n)L_3 \geq 2$ ;  $f(n)L_2 \geq 2$ . Table 2 lists the authentication solutions based on the functions.

**Table 2.** Authentication solution

Levels	Feedback Combinations	Value of $f(n)L_N$
Level 4	4 at Level 4	4
	3 at Level 4	3
Level 3	3 at Level 3	3
	2 at Level 3	2
	1 at Level 3 and 2 at Level 4	2.5
Level 2	2 at Level 2	2
	1 at Level 2 and 1 at Level 3 and 1 at Level 4	2.2

- If there are still not enough positive flags after the third try of Level 2, the server will turn to Level 1, where stores information of the user. At this point, a certain number of life questions will be sent to the user, which prompt him to response with the answers. Both of the life questions and the answers will be encrypted during the transmission:

$$(1) \{\{ \text{Questions} \} PV_S\} PK_A; (2) \{\{ \text{Answers} \} PV_A\} PK_S$$

The user needs to utilize his mobile phone PIN to decrypt the message in order to access and answer the questions.

## 4 Framework Analysis

### G. Analysis Description

The goal of the analysis is to evaluate the performance of the framework. We decided to calculate the Time Duration for which a user needs to wait in order to be authenticated. To calculate the Time Duration, concepts of Response Time and Waiting Time are worth being introduced. Response Time is defined as the time between sending out the first message to helper and receiving the feedback from the very last helper. Waiting Time is the duration of each try launched by the authentication server. Because

there are 3 tries for each level, we assume the waiting time is at the same length for all of tries at each level. Accordingly, there are three situations in the framework for each user. First, a user can be authenticated within a single level. Second, a user is authenticated by joining different levels. Third, a user is authenticated by himself. To make the description clearer, we analyze each situation in the following subsections.

#### *H. The First Situation Analysis*

There are totally 3 scenarios of this situation: Level-4 scenario, level-3 scenario, and level-2 scenario. We start the analysis from the lowest level. Level-4 has the majority issue that although four helpers are required to be picked, three feedbacks are enough for the authentication. It means that the Response Time for Level 4 is the time between sending out the first message to helper and receiving the third feedback. We also assume that a valid feedback should be always received within the Waiting Time. Therefore, the Response Time is always shorter than Waiting Time. For the level-4 scenario, there are ten cases. In the following, we describe each case.

- ◆ Case 1: Three feedbacks are collected within the first try.
- ◆ Case 2: Two feedbacks are collected after the first try and one feedback is collected within the second try.
- ◆ Case 3: One feedback is collected after the first try and two feedbacks are collected within the second try.
- ◆ Case 4: No feedback is collected after the first try and three feedbacks are collected within the second try.
- ◆ Case 5: Three feedbacks are collected within three tries, respectively.
- ◆ Case 6: Two feedbacks are collected after the first try. No feedback is collected within the second try and one feedback is collected within the third try.
- ◆ Case 7: One feedback is collected after the first try. No feedback is collected within the second try and two feedbacks are collected within the third try.
- ◆ Case 8: No feedback is collected after the first try. Two feedbacks are collected within the second try and one feedback is collected within the third try.
- ◆ Case 9: No feedback is collected after the first try. One feedback is collected within the second try and two feedbacks are collected within the third try.
- ◆ Case 10: No feedback is collected after the first two tries and three feedbacks are collected within the third try.

In general, these ten cases can be summarized by three categories:

1. Case 1: Three feedbacks are collected within first try.

$$T = T_{RES}$$

2. Case 2, 3, and 4: Three feedbacks are collected within first two tries.

$$T = T_{WAIT} + T_{RES}$$

3. Case 5, 6, 7, 8, 9, and 10: Three feedbacks are collected within three tries.

$$T = 2 * T_{WAIT} + T_{RES}$$

We use “T” to denote “Time Duration”, while “ $T_{RES}$ ” and “ $T_{WAIT}$ ” stand for Response Time and Waiting Time, respectively. The same case descriptions are applied

for level-3 and level-2 scenarios. The majority vote for level-3 still exists where two valid feedbacks are enough. For level-2, there is no majority vote consideration. Two valid feedbacks are required. This will result in the resemblance of level-3 and level-2 cases. Thus, we obtain the following five cases.

- ◆ Case 1: Two feedbacks are collected within the first try.
- ◆ Case 2: One feedback is collected after the first try and one feedback is collected within the second try.
- ◆ Case 3: No feedback is collected after the first try and two feedbacks are collected within the second try.
- ◆ Case 4: One feedback is collected after the first try. No feedback is collected within the second try and one feedback is collected within the third try.
- ◆ Case 5: No feedback is collected after the first two tries and two feedbacks are collected within the third try.

The five cases can be summarized by three categories.

Case 1: Two feedbacks are collected within first try.

$$T = T_{RES}$$

Case 2 and 3: Two feedbacks are collected within first two tries.

$$T = T_{WAIT} + T_{RES}$$

Case 4 and 5: Two feedbacks are collected within three tries.

$$T = 2 * T_{WAIT} + T_{RES}$$

In this first situation, all of the level scenarios share the same set of equations on Time Duration:  $T = T_{RES}$ ,  $T = T_{WAIT} + T_{RES}$ , and  $T = 2 * T_{WAIT} + T_{RES}$ . For further analysis, we conclude the set of equations in the following two equations:

- ◆  $T_1 = T_{RES}$
- ◆  $T_2 = k * T_{WAIT} + T_{RES}$ , where  $k = 1$  or  $2$

### *I. The Second Situation Analysis*

The second situation comes into reality when a user cannot be authenticated under one single level. The server calculates the joint feedbacks from certain levels to authenticate the user. In our framework, it happens only if the server is configured to choose helpers from Level 4. According to Table 4, there are two cases for this situation:

Case 1: One feedback from Level 3 and two feedbacks from Level 4.

Case 2: One feedback from each of Level 2, Level 3, and Level 4.

Time Durations of the two cases depend on  $T$  of Level 3 and Level 2 in sequence. Based on the conclusion of the first situation analysis, the Time Duration of case 1 are in two equations:

- 1)  $T = 3 * T_{WAIT} + T_1$
- 2)  $T = 3 * T_{WAIT} + T_2$

Similarly, the Time Duration of case 2 also has two equations:

- 1)  $T = 6 * T_{WAIT} + T_1$
- 2)  $T = 6 * T_{WAIT} + T_2$

For the further analysis, we document case 1's two equations as one equation of  $T_3$  and case 2's two equations as one equation of  $T_4$ . They are:

- ◆  $T_3 = m * T_{WAIT} + T_{RES}$ , where  $m = 3, 4$ , or  $5$
- ◆  $T_4 = n * T_{WAIT} + T_{RES}$ , where  $n = 6, 7$ , or  $8$

#### *J. The Third Situation Analysis*

In our framework, this situation occurs due to two reasons: (1) the user directly set the server to choose person from level-1; (2) there are not enough feedbacks after 3 tries at each helper level. Correspondingly, the Time Duration derived from the first reason is negligible since the user will receive the life questions immediately after he initiated the authentication process. For the second reason, it truly relies on the level from which the server is configured to choose helpers. Since we assumed that the server would start the authentication process from the lowest level, the Time Duration is  $T_5 = 9 * T_{WAIT}$ .

Notice that, even after  $T_5$ , there is still a period of time before the user is really authenticated. That is because he needs to read, answer, and submit the answered questions as in other situations the user needs to type and submit the PIN. To simplify, we assume this type of duration is negligible in the framework analysis.

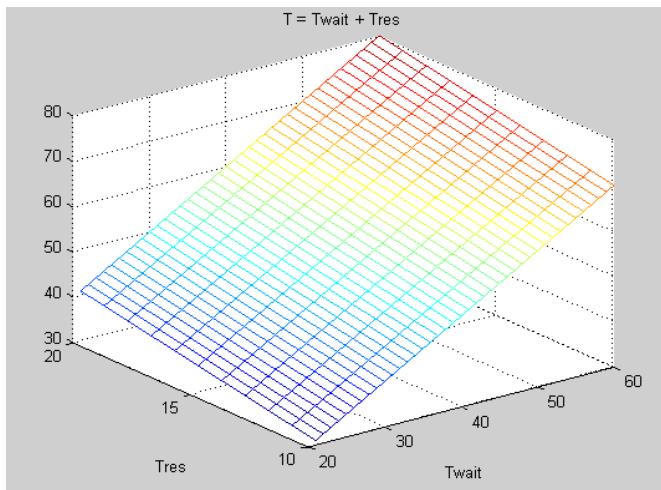
#### *K. Aggregation Analysis*

Five equations of Time Duration, which are taken into account for all the situations, have been built up through former analysis. To list all of them, the equations are:

- ◆  $T_1 = T_{RES}$
- ◆  $T_2 = k * T_{WAIT} + T_{RES}$ , where  $k = 1$  or  $2$
- ◆  $T_3 = m * T_{WAIT} + T_{RES}$ , where  $m = 3, 4$ , or  $5$
- ◆  $T_4 = n * T_{WAIT} + T_{RES}$ , where  $n = 6, 7$ , or  $8$
- ◆  $T_5 = 9 * T_{WAIT}$

To take an in-depth quantitative analysis, the value of  $T_{RES}$  is required. We have conducted simulation experiments within our national center to capture  $T_{RES}$  by calculating the time taken by a person to open, read, and reply a brief SMS message. The consequence shows that the range of  $T_{RES}$  is from 10 to 20 seconds depending on different persons. Since  $T_{WAIT}$  should be longer than  $T_{RES}$ , we assign a minimum integer value to  $T_{WAIT}$  which is 21. For its maximum value, we argue that the duration for each try is no longer than 60 seconds. In this case, we confirmed the interval of  $T_{RES}$  is [10, 20] and the interval of  $T_{WAIT}$  is [21, 60]. For the convenience,  $T_1, T_2, T_3$ , and  $T_4$  can be aggregated into one equation, which is  $T_A = A * T_{WAIT} + T_{RES}$ , where  $A = 1, 2, 3, 4, 5, 6, 7$ , or  $8$ . Although the aggregation may be perplexed, it holds a pragmatic meaning that  $T_A$  is a full collection of the Time Duration algorithms for helper-levels' authentication in the framework.

Figures 2 demonstrates the growing trends of T under certain “A” value within both intervals of  $T_{\text{WAIT}}$  and  $T_{\text{RES}}$ . With the different numbers of tries,  $T_A$  generates different minimum and maximum value pairs. Correspondingly, the value pairs can be found on each of the figures. Recall the equation of  $T_A$ , when the constant “A” equals zero, the consequence is  $T_{\text{RES}}$ , which indicates that the ultimate minimum value of  $T_A$  is 10 seconds. As “A” reaches its vertex value 8, the ultimate maximum value of  $T_A$  can be computed as 500 seconds. In this case, we conclude that the time duration for one user to be authenticated by helpers in our framework should be no longer than 500 seconds (8 minutes and 20 seconds). For the worst situation that all of helpers fail to authenticate the user, he only needs to wait, at most, 9 minutes to answer life questions to authenticate himself.



**Fig. 2.**  $T = T_{\text{WAIT}} + T_{\text{RES}}$

## 5 Discussion

The magnitude of T relies on the value of two variables,  $T_{\text{RES}} \in [10, 20]$  and  $T_{\text{WAIT}} \in [21, 60]$ . It is true that  $T_{\text{WAIT}}$  should be always longer than  $T_{\text{RES}}$  in order to supply valid feedbacks. However, it might be inefficient to maintain a fixed  $T_{\text{WAIT}}$  for each helper by using such an interval, since 60 seconds is still three times of 20 seconds, the upper boundary of  $T_{\text{RES}}$ . In this case, the goal of this analysis becomes that finding the minimum  $T_{\text{WAIT}}$  to minimize T.

We assume that for each  $T_{\text{RES}}$ , there is a minimum  $T_{\text{WAIT}}$ , which equals  $T_{\text{RES}} + C$ , where C is the time duration after a feedback is sent out but before it can be received and verified by the authentication server. To test the value of C, we simulate this situation by sending a SMS message. Given the reason that a feedback needs to be decrypted before the server verifies [11], we decided to expand C to 5 seconds. Hence, a new relationship between  $T_{\text{WAIT}}$  and  $T_{\text{RES}}$  is built up.

$$T_{\text{WAIT}} = T_{\text{RES}} + 5.$$

Accordingly, the equation of  $T_A$  is transformed to

$$T = A * (T_{RES} + 5) + T_{RES}.$$

Figure 3 demonstrates the eight trends of  $T$  with  $A = 1, 2, 3, 4, 5, 6, 7$ , and  $8$ , respectively. Comparing each trend's maximum value with its counterpart, a significant maximum value drop is identified after the relationship  $T_{WAIT} = T_{RES} + 5$  is built. For instance, as  $A$  equals to  $8$ , the maximum value of  $T$  is  $220$  seconds. It is more than two times lower than  $500$  seconds, which is the ultimate maximum value of  $T_A$ . Therefore, the goal of minimizing the time duration is achieved.

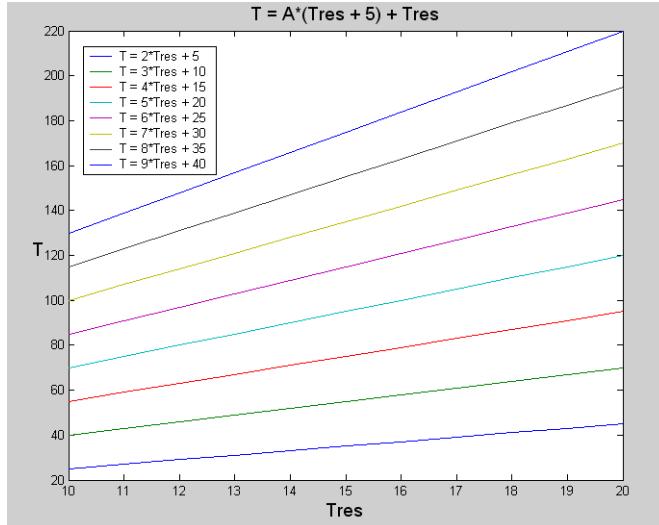


Fig. 3.  $T = A*(T_{RES} + 5) + T_{RES}$

## 6 Security and Attack Analysis

There are two types of potential attacks towards our framework: (1) attack towards users, and (2) attack towards helpers. Let us use  $(X(u):VA)$  to indicate an event which a party  $X$  claiming as identity  $u$  and correctly presenting the value item of  $VA$ .

### L. Attack Towards Users

A user ( $U$ ) is subject to be attacked because an attacker ( $A$ ) has incentives to access  $U$ 's account. Hence,  $A$  impersonates  $U$  ( $A(u)$ ) and conducts this type of attack in the following two ways:

- 1) A maliciously initiates the authentication process. The challenges of this attack are described as:
  - a)  $A$  needs to obtain  $U$ 's mobile phone ( $A(u):MP$ ).
  - b)  $A$  needs to acquire  $U$ 's username ( $A(u):UserName$ ), password ( $A(u):PassWord$ ), and mobile phone PIN ( $A(u):PIN$ ) in order to initiate the attack.

- 2) A attacks U after the latter one got one temperoray token code. The challenges are:
- A needs to obtain the token code ( $A(u):TK$ ).
  - A needs to acquire U's username ( $A(u):UserName$ ) , password ( $A(u):PassWord$ ) so that to access U's account together with the token code.

We also assume that the probabilities of all the events are independent. Events ( $A(u):PassWord$ ), ( $A(u):PIN$ ), and ( $A(u):TK$ )are small probability events, where  $\{A(u):PassWord\} < \varepsilon$ ,  $prob\{A(u):PIN\} < \delta$  and  $prob\{A(u):TK\} < \mu$ .

An accomplished attack, ( $A(u):AccountAccess$ ), is a total overcome of all the challenges. The probability of a successful attack is described as:

$$\begin{aligned} prob\{A(u):AccountAccess \text{ in 1}\} \\ = prob\{A(u):MP\} \cap prob\{A(u):UserName\} \\ \cap prob\{A(u):PassWord\} \cap prob\{A(u):PIN\} \\ \\ prob\{A(u):AccountAccess \text{ in 2}\} \\ = prob\{A(u):TK\} \cap prob\{A(u):UserName\} \\ \cap prob\{A(u):PassWord\} \end{aligned}$$

Given that U may not treat his mobile phone and username as confidential as the password and PIN, therefore, ( $A(u):MP$ ) and ( $A(u):UserName$ ) will not be small probability events. Considering the worst case, where  $prob\{A(u):MP\} = prob\{A(u):UserName\} = 1$ , the probability is computed as:

$$\begin{aligned} prob\{A(u):AccountAccess \text{ in 1}\} \\ = 1 * 1 * \varepsilon * \delta = \varepsilon * \delta \\ prob\{A(u):AccountAccess \text{ in 2}\} \\ = \mu * 1 * \varepsilon = \mu * \varepsilon \end{aligned}$$

Both  $\varepsilon * \delta$  and  $\mu * \varepsilon$  are two small probabilities indicating that A has very little chance to accomplish the attack. Even if A has overcome all of the challenges and ready to conduct a successful attack, the last security line is that the server allows U to disable the social authentication service through configuring the authentication webpage settings. Once the service has been disabled, it requires U to enable it again by personally visiting the agency. This gives U a chance to survive from the disaster prior to an attack. A successful attack through either of the two ways is difficult. At first, obtaining the mobile phone is difficult if A is a stranger to U. It is almost impossible for A to stealthily use the phone under this situation. Stealing it may alarm U and force him to disable the service timely. Since A is an acquaintance of U, stealthily utilizing the phone actually depends on a coincidence that the mobile phone is currently out of U's sight or control. Other than that, A still needs to figure out how to get U's mobile phone PIN, username, and password. If having an access to any of the three secrets can be considered as a small probability event, successfully collecting all the credentials is based on a much smaller probability which is a product of the three small probability events.

#### *M. Attack Towards Helper*

The other type of attack is an attacker (A) attacks a helper. One purpose to conduct this type of attack is to make U authenticate himself by answering the life questions then

enhancing the probability of answer disclosure. Therefore if the answer is able to be acquired as well as valuable to A, U may suffer from other insidious incidents. There are enormous challenges to launch this type of attack since A should be able to (1) have the knowledge of the helpers on that multi-level list; (2) make the social authentication service unavailable by stealing multiple phones from helpers; (3) let U conduct the social authentication service, for example, sabotaging U's token device; and (4) get access to the answers of the life questions. Generally, a multi-level list's contents should be only known by U. The list is securely stored on the authentication server. If A wants to know the list, the best bet for A is to conduct social engineering on U. Stealing multiple phones from helpers to make the social authentication service unavailable is extremely difficult since it requires A to steal at least 25 phones according to our current example scheme. Forcing U to conduct social authentication indicates that U does not have access to a fresh token code. A can achieve this by stealing or sabotaging U's token device. However, both of the behaviors are considered difficult. When U types answers of the life questions into his mobile phone, a shoulder-surfing attack is able to allow A to collect the answers but this type of attack can be prevented [11].

#### *N. Man-in-the-Browser Attack Analysis*

A Man-in-the-Browser attack [13] is able to secretly intercept as well as tamper the framework communications by inserting or deleting the content of the communications. This attack applies to our framework. In order to quench the fire, it is able to impose a prevention scheme onto the Mobile Phone Server that as the server contacts with helpers it also sends a copy of the contact information to user. In this case, the user should be able to receive the copy of contact information after she initiates the authentication process. A failure of receiving the information will then alarm the user.

## 7 Conclusions

In this paper, we develop an authentication framework using multi-level social networks to automate the social authentication process. To evaluate the performance of the framework, we conduct a thorough framework analysis which is divided into several sub-analysis with a unique goal that is to calculate the time duration of the framework. Asymmetric encryption algorithms are applied during the whole process of the framework to safeguard confidentiality, integrity, and availability of the message. Besides, we described attack analysis of the framework that the successful attacks are difficult to be achieved due to the multiple challenges. As a future work, we intend to fully implement the authentication framework.

## References

1. Brainard, J., Juels, A., Rivest, R., Szydlo, M., Yung, M.: Fourth-Factor Authentication: Somebody You Know. In: Proceedings of the 13th ACM Conference on Computer and Communications Security (October 2006)
2. Soleymani, B., Maheswaran, M.: Social Authentication Protocol for Mobile Phones. In: International Conference on Computational Science and Engineering, Vancouver, Canada (August 2009)

3. Podd, J., Bunnell, J., Henderson, R.: Cost-effective Computer Security: Cognitive and Associative Passwords. In: Proceedings of the 6th Australian Conference on Computer-Human Interaction, Washington, DC, USA, p. 304 (1996)
4. Zviran, M., Haga, W.: User Authentication by Cognitive Passwords: An Empirical Assessment. In: Proceedings of the 5th Jerusalem Conference on Information Technology, Los Alamitos, CA, USA, pp. 137–144 (1990)
5. Ellison, C.: UPnP Security Ceremonies Design Document: For UPnP Device Architecture 1.0 (October 2003),  
[http://www.upnp.org/download/standardizeddcps/  
UPnPSecurityCeremonies\\_1\\_0secure.pdf](http://www.upnp.org/download/standardizeddcps/UPnPSecurityCeremonies_1_0secure.pdf)
6. McCune, J., Perrig, A., Reiter, M.: Seeing-Is-Believing: Using Camera Phones For Human-Verifiable Authentication. In: IEEE Symposium on Security and Privacy, Berkeley/Oakland, California, pp. 110–124 (May 2005)
7. Schechter, S., Egelman, S., Reeder, R.: It's Not What You Know, But Who You Know: A Social Approach To Last-Resort Authentication. In: Proceeding of the 27th Annual SIGCHI Conference on Human Factors in Computing Systems, Boston, MA, USA (April 2009)
8. Yardi, S., Feamster, N., Bruckman, A.: Photo-Based Authentication Using Social Networks. In: Proceedings of the 1st Workshop on Online Social Networks, Seattle, Washington, USA (August 2008)
9. Aloul, F., Zahidi, S., EI-Hajj, W.: Two Factor Authentication Using Mobile Phones. In: IEEE/ACS International Conference on Computer Systems and Applications, Rabat, Morocco (May 2009)
10. Trusted Computing Group, <http://www.trustedcomputinggroup.org/>
11. Mannan, M., Van Oorschot, P.C.: Using a personal device to strengthen password authentication from an untrusted computer. In: Dietrich, S., Dhamija, R. (eds.) FC 2007 and USEC 2007. LNCS, vol. 4886, pp. 88–103. Springer, Heidelberg (2007)
12. Santesson, S., Housley, R., Bajaj, S., Rosenthal, L.: Internet X.509 Public Key Infrastructure - Certificate Image, Internet Engineering Task Force (November 2009)
13. Guhring, P.: Concepts Against Man-in-the-Browser Attacks (2006),  
[http://www2.futureware.at/svnlsourcerer/  
CAcert1SecureClient.pdf](http://www2.futureware.at/svnlsourcerer/CAcert1SecureClient.pdf)

**PART I**

**Knowledge Discovery and**

**Information Retrieval**

# Extracting Relationship Associations from Semantic Graphs in Life Sciences

Weisen Guo and Steven B. Kraines

Science Integration Program (Human)

Department of Frontier Sciences and Science Integration, Division of Project Coordination  
The University of Tokyo, 5-1-5 Kashiwa-no-ha, Kashiwa, Chiba, 277-8568, Japan  
`{gws, sk}@scint.dpc.u-tokyo.ac.jp`

**Abstract.** The rate of literature publication in life sciences is growing fast, and researchers in the bioinformatics and knowledge discovery fields have been studying how to use the existing literature to discover novel knowledge or generate novel hypothesis. Existing literature-based discovery methods and tools use text-mining techniques to extract non-specified relationships between two concepts. This paper presents a new approach to literature-based discovery, which adopts semantic web techniques to measure the relevance between two relationships with specified types that involve a particular entity. We extract pairs of highly relevant relationships, which we call relationship associations, from semantic graphs representing scientific papers. These relationship associations can be used to help researchers generate scientific hypotheses or create their own semantic graphs for their papers. We present the results of experiments for extracting relationship associations from 392 semantic graphs representing MEDLINE papers.

**Keywords:** Relationship associations, Semantic relationships, Semantic matching, Semantic web, Semantic graph, Life sciences, Literature-based knowledge discovery.

## 1 Introduction

The field of life sciences is one of the fastest growing academic disciplines [1]. More than one million papers are published each year in a wide range of biology and medicine journals [2]. Recent progress in genomics and proteomics has generated large volumes of data on the expression, function, and interactions of gene products. As a result, there is an overwhelming amount of experimental data and published scientific information, much of which is available online.

Scientific discovery is a type of human intellectual activity. Based on observations and theory, researchers define hypotheses that they test experimentally. However, due to the explosive growth of the literature, individual scientists cannot study all of the experimental data and scientific information that is available.

Researchers in the bioinformatics and knowledge discovery fields have been studying how to apply computational methods to the existing literature to discover novel

knowledge or generate novel hypotheses [3], [4]. For example, informatics tools have been developed that replicate Swanson's discovery in 1986 from analysis of the literature that fish oil may benefit patients with Raynaud's disease [5], [6], [7], [8]. The process of linking different scientific disciplines through intermediate, or shared, topics in the literature has been termed "literature-based knowledge discovery." Most of the existing work on literature-based knowledge discovery employ text-mining techniques to find relationships of unspecified type between two domain-related concepts that are implied by relationships with a third common concept existing the literature, which is often described as Swanson's ABC model.

In this paper, we present a technique for literature-based discovery of hypotheses by measuring the association between two relationships of specified type that involve a common entity or concept. We call this a "relationship association": a special kind of association rule that states "if concept A has relationship R1 with concept B, then it is likely that concept A has relationship R2 with concept C." Note that the emphasis is on the association between the relationships ( $A \rightarrow R1 \rightarrow B$ ) and ( $A \rightarrow R2 \rightarrow C$ ), not the single concepts A, B and C. The rationale for using this approach to hypothesis generation is as follows.

It has been noted that while knowledge in domains of practice such as health care can often be expressed as single concepts, most scientific knowledge takes the form of relationships between concepts from the study domain, which have been identified through research [9]. A relationship represents a statement that predicates the way in which one concept modifies the other semantically. Our goal is to discover interesting association rules between these relationships.

Unfortunately, current text mining techniques cannot extract relationships between concepts with semantics that are sufficiently precise for this kind of analysis [10]. We use semantic web techniques and ontologies to define semantic relationships described in a scientific paper as follows. First, we create a descriptor for each paper in the form of a semantic graph. The nodes in a semantic graph consist of instances of particular concepts defined in the ontology that represent entities described in the paper. The edges in a semantic graph are the specific relationships that the paper describes between those entities (an example is shown in Fig. 1). For example, the statement "a **Flagellum** called *chlamydomonas flagellum* has as a structure part a **Cytoskeleton** called *axoneme*" (here bold type indicates class terms and italic type indicates instance terms as described below) is a relationship represented by one arc in the semantic graph shown in Fig. 1. Then, all pairs of relationships from the semantic graphs that share a common entity, e.g. all chains with three nodes and two arcs, are candidates for relationship associations.

We envisage two primary usages of relationship associations. One is helping biological scientists to generate novel hypotheses. For example, the relationship association that "if some kind of cellular structure is part of some kind of flagellum, then it is likely that the cellular structure binds to a specific biological entity" might inspire a biologist studying a particular kind of cellular structure, such as a microtubule, that is part of a flagellum to hypothesize that the cellular structure binds to some other biological entity in the studied cell. Another possible application of relationship associations is to help users to create computer-interpretable descriptors of their papers in some knowledge sharing system, such as EKOSS [11]. For example, when the user creates a relationship describing how one instance is modified by another, and this

relationship appears in one part of a relationship association, then the system could automatically suggest a new relationship and target instance to add to the instance based on the other part of the association.

Our approach has two basic assumptions. First, because relationship associations describe associations of relationships between classes of entities, we assume that similar entities have similar relationships. Second, because we use semantic graphs from a small part of the scientific literature to extract the relationship associations, we assume that if one relationship association appears in the sample data with a high probability, then it will also appear in the whole literature with a similar probability.

This paper is organized as follows. In Section 2, we describe our work that forms the background for this paper. In Section 3, we present our approach to extract the relationship associations. In Section 4, we describe experiments using 392 semantic graphs for papers from MEDLINE to obtain relationship associations. The presentation and experimental application of the algorithm for extracting relationship associations are the main contributions of this paper. In Section 5, we discuss related work. In the final section we describe some of the future work we are doing.

## 2 Background

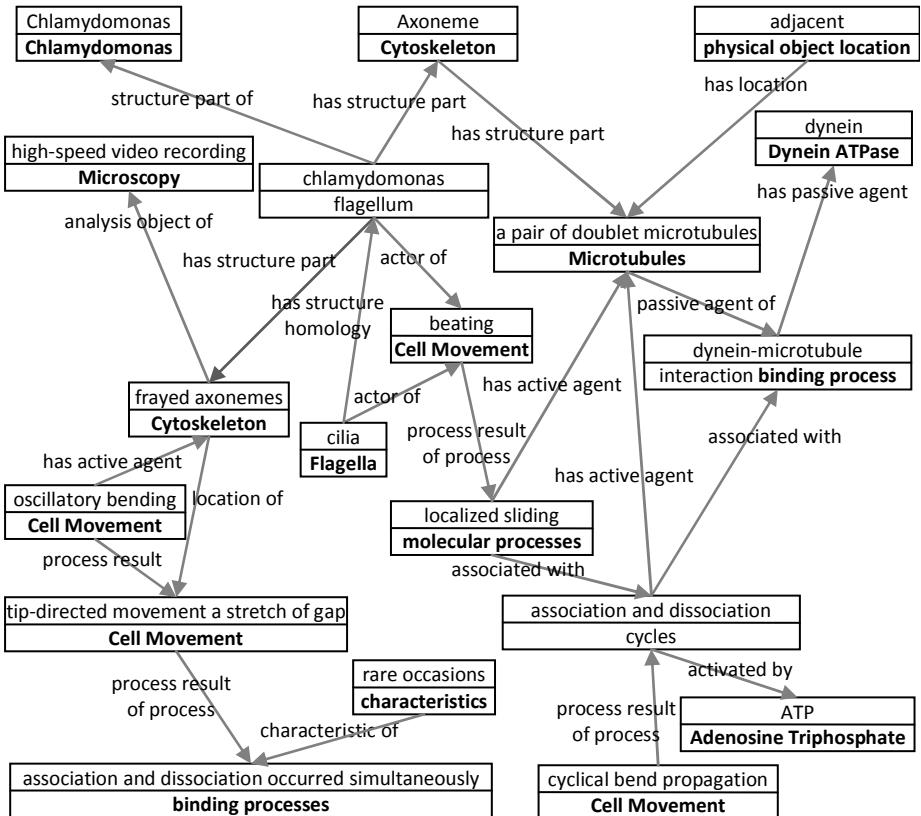
Many applications of semantic web technologies in the life sciences have appeared recently, including several large ontologies for annotating scientific abstracts, such as the Open Biomedical Ontologies (OBO) and the Unified Medical Language System (UMLS) Semantic Network. We have employed several of these technologies in developing the relationship association extraction technique.

### 2.1 Creation of Semantic Graphs

EKOSS (Expert Knowledge Ontology-based Semantic Search) [11] is a web-based knowledge-sharing system that enables users to create one or more semantic graphs describing their knowledge resources, such as scientific papers, using ontologies. EKOSS then uses a reasoner to match a semantic query against the semantic graphs. In previous work, we have used EKOSS to create semantic graphs for 392 papers selected from MEDLINE. We use one semantic graph to represent one MEDLINE paper. In order to describe a paper from MEDLINE as a semantic graph, we developed the UoT ontology based on a subset of the Medical Subject Headings (MeSH) vocabulary [12]. The nodes of a semantic graph are instances of ontology classes, and the edges are relationships between the instances that are specified by properties also defined in the ontology. Each instance can have a descriptive text label. Fig. 1 shows a semantic graph created to describe a paper from MEDLINE [13]. The semantic graph contains 19 instances of classes from the UoT ontology together with 23 relationships between the instances.

### 2.2 Semantic Matching

Unlike text matching, which is usually based on calculating the similarity of two strings [14], semantic matching techniques compare two data structures at a semantic level, often by using some logic inference methods.



**Fig. 1.** The semantic graph of a paper from MEDLINE [13]. Boxes show instances of classes from the domain ontology. The text above the line in a box is the instance label. The text in bold type below the line in a box is the class name of that instance. Arrows show properties expressing the asserted relationships between instances.

We use the description logics reasoner, RacerPro [15], to evaluate the match between a search semantic graph and a target semantic graph through the following process. First, we add the target graph to the reasoner's knowledge base together with the ontology used to create the graph. Then, we convert the search graph into a set of semantic queries by creating sub graphs of the search graph that contain a specified number of properties and instances. For the work presented here, we create queries with one property and two instances, i.e. a semantic triple. Queries are created by replacing the instances in the sub graphs with class variables. Rules for replacing instance classes with super classes and properties with super properties can be applied to increase matching recall. Finally, we use the reasoner to evaluate how many of the queries match the target graph, where a query matches if the reasoner can find instances in the target graph that can be bound to each of the class variables in the query subject to the specified relationship(s). The fraction of matching queries gives the

semantic similarity between the two graphs. A simple example is shown in Fig. 2 (in 3.2). Details are given in [16].

### 3 Relationship Association Extraction

The process of extracting relationship associations involves three parts: the data structure, the method for determining if a relationship association appears in a particular semantic graph, and the algorithm for extracting the relationship associations from a set of semantic graphs. We use the semantic graph (in 2.1) as the data structure to represent the papers and the semantic matching technique (in 2.2) as the method for determining if a relationship association appears in a particular semantic graph. From the set of semantic graphs, we generate a set of linked pairs of semantic relationships, where each relationship is defined as a triple consisting of a subject or “domain” class, an object or “range” class, and a directed property specifying the relationship between the two classes. A linked pair of semantic relationships is a pair of semantic relationships that share one class in common. We refer to these linked pairs of semantic relationships as relationship associations.

#### 3.1 Generating Triple Queries

A semantic triple – consisting of a domain instance, a range instance, and a property between them – is the minimum unit of a semantic graph. One semantic graph contains one or more semantic triples. The definitions are formalized as follows:

$$\text{Graph} = \{\text{Triple}^*\}$$

$$\text{Triple} = \{\text{domain}, \text{property}, \text{range}\}$$

For each triple in a semantic graph, we create one triple query, defined as follows:

$$\text{TripleQuery} = \{\text{domain class variable}, \text{property}, \text{range class variable}\}$$

In a triple query the instances of the triple are converted to variables with the same classes. Thus, a triple query converts the asserted relationship between two specific entities made by the triple into a generalized relationship between ontology classes.

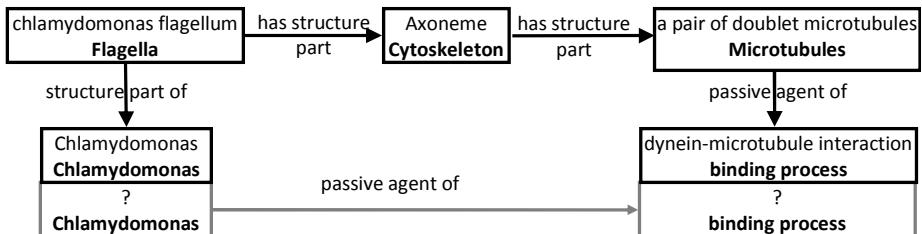
There may be some duplicate triple queries generated from the set of semantic graphs. However, because we only want to link two triple queries whose triples both appear in the same semantic graph and share a common entity, we keep all of the generated triple queries at this point.

#### 3.2 Matching Triple Queries

We use RacerPro to infer matches between queries and graphs via both logical and rule-based reasoning. The logic is built into the ontology using formalisms provided by the description logic that is supported by the ontology specification we used (OWL-DL [17]). The rules are pre-defined for a particular ontology by domain experts. Details are given in [11], [16].

If RacerPro can find a pair of instances in a particular semantic graph meeting the class and relationship constraints of a triple query *Query1*, then we say that the triple *Triple1* represented by *Query1* appears in the semantic graph. By using both logical

and rule-based reasoning, we can get matching results that are implied at a semantic level because the reasoner can infer relationships between instances that are not explicitly stated in the semantic graph. For example, consider the segment of the semantic graph in Fig. 1 between the instance of the class **Chlamydomonas** called *Chlamydomonas* and the instance of the class **binding process** called *dynein-microtubule interaction*. The triple query “find some instance of **Chlamydomonas** that is a passive agent of some instance of **binding process**” does not actually occur in the graph because there is no property between *Chlamydomonas* and *dynein-microtubule interaction*. However, Fig. 2 shows that the query matches with the semantic graph because the relationship is implied by the relationships specified with the instance of the class **Flagella** called *chlamydomonas flagellum*, the instance of the class **Cytoskeleton** called *frayed axonemes*, and the instance of the class **Microtubules** called *a pair of doublet microtubules*. This match is a result of the rule “If A has structure part B and B is passive agent of C, then A is passive agent of C” together with the transitivity of the “has structure part” relationship, and the inverse relationship between the “has structure part” and “structure part of”.



**Fig. 2.** An example of semantic matching. Instances are indicated with boxes where the first line of text gives the instance name and the second line gives the instance class. Properties are shown by directed arrows labelled with the property name. The part in outlined in black is from the semantic graph. The part in outlined in gray is the query.

Using RacerPro, we match all triple queries with all semantic graphs and count the number of graphs in which each triple query occurs. If the triple query just occurs in the one semantic graph from which the triple was obtained, then it cannot give more information for the extraction of relationship association and it is removed. The remaining triple queries are used to create association queries in the next step.

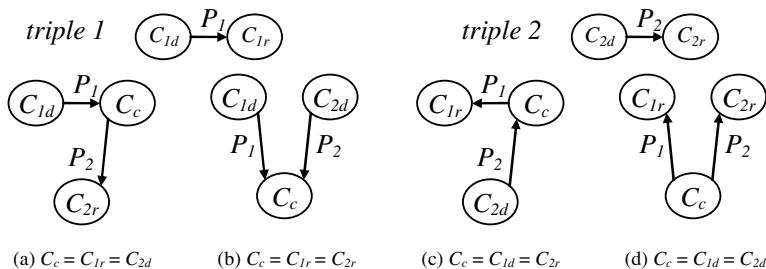
### 3.3 Generating Association Queries

For each graph, we find all pairs of triples that share one instance and therefore comprise a connected segment of the semantic graph having three instances and two properties. If both of the corresponding triple queries are in the set of triple queries generated in 3.2, then we create an association query corresponding to the connected segment containing the two triples. However, this can result in association queries that have the same semantic interpretation. To decrease the computational load of matching the association queries with the set of graphs, we remove the duplicates in the next step.

### 3.4 Removing Duplicate Queries

Because we use semantic matching to match an association query with a semantic graph, two queries that are semantically equivalent will get the same matching results. By removing such duplicate association queries, we can reduce the number of reasoning steps, a computationally expensive task, that must be performed.

The graphs are directed, so even for two queries having the same classes and properties, if the directions of the properties are different, then the queries may be different. Therefore, we must consider four types of association queries as shown in Fig. 3. One association query is comprised of two triple queries that share one connecting class.  $C_{1d}$ ,  $C_{1r}$  and  $P_1$  are the domain class, the property and the range class of the first triple query, respectively.  $C_{2d}$ ,  $C_{2r}$  and  $P_2$  are the domain class, the property and the range class of the second triple query, respectively.  $C_c$  represents the connecting class of the two triple queries.



**Fig. 3.** Four types of association queries

We describe the process of removing duplicate association queries for the case where the original query,  $Q1$ , is type (a) (the processes for queries of type (b), (c), and (d) are similar). We append a subscript “1” to the notation shown in Fig. 3 to indicate classes and properties in  $Q1$ .

$$Q1 = \{C_{1d1} \rightarrow P_{11} \rightarrow C_{c1}; C_{c1} \rightarrow P_{21} \rightarrow C_{2r1}\}$$

We want to determine if a second association query,  $Q2$ , has the same semantic interpretation. If  $Q2$  also is type (a):

$$Q2 = \{C_{1d2} \rightarrow P_{12} \rightarrow C_{c2}; C_{c2} \rightarrow P_{22} \rightarrow C_{2r2}\}$$

and meets the following conditions:

$$(C_{1d1} = C_{1d2}; P_{11} = P_{12}; C_{c1} = C_{c2}; P_{21} = P_{22}; C_{2r1} = C_{2r2})$$

then  $Q2$  is an exact match with  $Q1$  and so we remove it.

If not, we consider symmetric properties and inverse properties. For example, if there are two triples “ $A \rightarrow P_1 \rightarrow B$ ” and “ $B \rightarrow P_2 \rightarrow A$ ” and  $P_1$  is the inverse of  $P_2$ , then the triples have the same meaning. If at least one property in  $Q2$  is symmetric or has an inverse property, we use the following algorithm to create additional association queries having the same semantic interpretation as  $Q2$ . If any of the new association queries matches with  $Q1$  using the process above, we remove  $Q2$ .

```

Algorithm Create semantically equivalent queries
triple1 = ( $C_{1d} \rightarrow P_1 \rightarrow C_{1r}$ )
triple2 = ( $C_{2d} \rightarrow P_2 \rightarrow C_{2r}$ )
newQueries = {}
triple1Candidates = {triple1}
triple2Candidates = {triple2}
If  $P_1$  is inverse, Then
    triple1Candidates += ( $C_{1r} \rightarrow \text{INV}(P_1) \rightarrow C_{1d}$ )
Else If  $P_1$  is Symmetric, Then
    triple1Candidates += ( $C_{1r} \rightarrow P_1 \rightarrow C_{1d}$ )
End If
If  $P_2$  is inverse, Then
    triple2Candidates += ( $C_{2r} \rightarrow \text{INV}(P_2) \rightarrow C_{2d}$ )
Else If  $P_2$  is Symmetric, Then
    triple2Candidates += ( $C_{2r} \rightarrow P_2 \rightarrow C_{2d}$ )
End If
For each  $tn1$  in triple1Candidates
    For each  $tn2$  in triple2Candidates
        If  $tn1 \neq \text{triple1}$  or  $tn2 \neq \text{triple2}$ , Then
            newQueries += combine ( $tn1, tn2$ )
        End If
    End For
End For
Return newQueries

```

### 3.5 Matching Association Queries

The matching technique described in Step 3.2 is used to match the association queries with each of the semantic graphs in the corpus and calculate the frequencies in which they occur. Association queries that just occur in one semantic graph are removed. The rest of the queries are candidates for relationship associations.

### 3.6 Relevance Criteria

From the previous steps, we get a set of association queries with unique semantic interpretations that occur in at least two semantic graphs. In order to help users find useful relationship associations, we consider two criteria for the relevance or meaningfulness of a relationship association. First, the first triple (the conditional triple) of the association rule should be relatively uncommon because if the conditional triple occurs in most of the graphs then it simply represents a general condition in the corpus. Second, the second triple should occur more often in combination with the first triple than in combination with connecting class. The rationale for the second criterion is that if the second triple occurs almost every time that the connecting class occurs, then the occurrence of the relationship association is just a consequence of the presence of the connecting class in the first triple. Thus, the association is not really between two relationships, but between a relationship (the first triple) and an entity (the connecting class).

We can calculate the second criterion from the frequencies of the association query  $P(a)$ , the two triples  $P(t1)$  and  $P(t2)$ , and the connecting class  $P(cc)$ . First, from Bayes' Theorem, we have  $P(t_2|t_1)=P(t_1|t_2) P(t_2) / P(t_1)$ . However,  $P(t_1|t_2) P(t_2)$  is just  $P(a)$ , so we

have  $P(t_2|t_1) = P(a)/P(t_1)$ . Next, from  $P(t_2|cc) = P(cc|t_2) P(t_2) / P(cc)$  and the observation that  $P(cc|t_2) = 1$ , we can write  $P(t_2|cc) = P(t_2) / P(cc)$ . Replacing the frequencies with supports, we have:

$$\begin{aligned} P(t_2|t_1)/P(t_2|cc) &= (P(a)/P(t_1)) / (P(t_2)/P(cc)) \\ &= (S_a/S_{t_1}) / (S_{t_2}/S_{cc}) \end{aligned}$$

where  $S_a$  is the support for the association query,  $S_{cc}$  is the support for the connecting class, and  $S_{t_1}$  and  $S_{t_2}$  are the supports for the respective triples obtained in Step 3.2.

Therefore, we select association queries meeting the following conditions:

$S_{t_1} \ll$  the size of corpus (the first triple is not so common)

$S_a/S_{t_1} \gg S_{t_2}/S_{cc}$  (the second triple occurs more often in combination with the first triple than in combination with connecting class)

### 3.7 Relationship Associations

As a result of the extraction process described above, we get a set of association queries filtered by relevance criteria together with their frequencies of occurrence. Because this information can be difficult for users to understand in graph form, we use templates and simple natural language generation algorithms to create natural language expressions of the relationship associations from the association queries [18]. These relationship associations can be further examined to identify those that are most reasonable and interesting. These final relationship associations can be used to generate scientific hypotheses or to help users to create new semantic graphs.

## 4 Experiments

Using the process described above, we have conducted experiments to obtain relationship associations from a set of 392 MEDLINE papers. In this section, we report the results of this experiment.

As described in section 2, semantic graphs were created for 392 papers selected from MEDLINE using the UoT ontology that we have developed in other work [12]. The UoT ontology has 1,762 classes and 151 properties, which we used to create 392 semantic graphs. The entire set of graphs contains 10,186 instances and 13,283 properties, so on average, each semantic graph has 26 instances and 34 properties.

We created 13,283 triple queries from the 392 semantic graphs and then used RacerPro to determine how many semantic graphs contain each triple. After removing all triple queries that only matched with the graph from which the triple was obtained, 8,200 triple queries remained that were available for creating association queries.

We created 18,704 association queries based on the 8,200 triple queries and 392 graphs. We removed duplicates using the method from 3.4. We also removed highly general queries. For example, the property “associated with” in UoT ontology is the top-level of the property hierarchy. Therefore, a query containing that property does not give us any information about the relationship type. Other highly general “stop list”

queries can be added as required. As a result, 4,821 association queries were obtained from the 392 semantic graphs.

We matched these association queries with all of the semantic graphs using Racer-Pro and removed all queries that only appeared once. This resulted in a total of 2,113 association queries appearing in at least two of the semantic graphs.

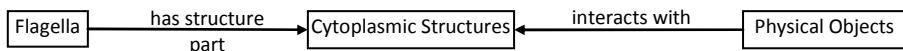
Next, we calculated the relevance criteria for these association queries. We used a cutoff value of  $S_{t1} \leq 40$  for the first criterion, which means that the first triple must occur in no more than 40 semantic graphs. We used a cutoff value of  $(S_d/S_{t1})/(S_{t2}/S_{cc}) \geq 2$  for the second criterion, which means that the probability that the association query occurs when the first triple occurs must be twice the probability that the second triple occurs when the connecting class occurs. A total of 984 association queries met these selection criteria, five of which are shown in Table 1.

**Table 1.** Five association queries and their selection criteria. Each triple is shown in the form “domain class | property | range class”. The conditional triple (*t1*) is separated from the consequent triple (*t2*) using “>”. The connecting class is shown in bold type.  $S_a$ ,  $S_{t1}$ ,  $S_{t2}$  and  $S_{cc}$  denote the numbers of semantic graphs in which the relationship association, triple *t1*, triple *t2*, and connecting class occur, respectively; *Criterion2* =  $(S_d/S_{t1})/(S_{t2}/S_{cc})$ .

Association query							
$S_a$	$S_{cc}$	$S_{t1}$	$S_{t2}$	$S_d/S_{t1}$	$S_{t2}/S_{cc}$	<i>Criterion2</i>	
Flagella   has structure part   <b>Cytoplasmic Structures</b> > physical objects   interacts with   <b>Cytoplasmic Structures</b>							
5	61	6	22	0.83	0.36	2	
<b>Cytoplasmic Structures</b>   has structure part   Microtubules > Chlamydomonas   has structure part   <b>Cytoplasmic Structures</b>							
4	61	7	5	0.57	0.08	7	
<b>Cells</b>   passive agent of   Neoplasms > Cell Proliferation   has active agent   <b>Cells</b>							
4	164	12	9	0.33	0.05	7	
<b>Gene Expression</b>   has passive agent   Receptors, Cell Surface > <b>Gene Expression</b>   has location   Neurons							
4	106	10	6	0.4	0.06	7	
<b>organism parts</b>   structure part of   Drosophila > Growth and Development   has passive agent   <b>organism parts</b>							
4	254	6	29	0.67	0.11	6	

Finally, we converted the association queries into natural language expressions, and we asked an expert in life sciences to identify the most interesting relationship associations. One example of a relationship association that was found to be interesting is shown in Fig. 4. The natural language representation is: “If a **Cytoplasmic Structure** is part of a **Flagellum**, then the probability that there is a **Physical Object** that interacts with the **Cytoplasmic Structure** is very high.” This relationship association is considered to be of at least some interest in life sciences because it indicates

that most studies of flagella in microorganisms which focus on the structures that make up the flagella also tend to focus on the interactions of the structures making up the flagella with other specified objects.



**Fig. 4.** An example of a relationship association

This relationship association appears in five papers in our experiment:

1. “Eukaryotic flagellum is a **Flagellum** that has as a part some **Cellular Structure** called *flagellar axoneme*. The *flagellar axoneme* has as a part some **Microtubule** called *doublet microtubule* that interacts with a **Dynein ATPase** called *dynein arms*.” [19] In this statement, the interaction is between the **Microtubule** (a **Cytoplasmic Structure**) and the **Dynein ATPase** (a **Physical Object**). The part of relationship between the **Microtubule** and the **Flagellum** is inferred from the transitivity of the “has structure part” relationship.
2. “There is a **Flagellum** that has as a part some **Cellular Structure** called *axoneme*. *Sliding disintegration* is a **molecular process** that consumes the *axoneme* and that is regulated by some **Ion** called *Ca(2+)*.” [20] In this statement, the **Cytoplasmic Structure** is the *axoneme* and the **Physical Object** is the *Ca(2+)*. The “interacts with” relationship is inferred using the rule that “if **Process P** acts on **Object A** and is regulated by **Object B**, then **Object A** interacts with **Object B**.”
3. “*Chlamydomonas flagellum* is a **Flagellum** that has as a part a **Cytoskeleton** called *axoneme*. The *axoneme* has as a part some **Microtubule** called *a pair of doublet microtubules* that participates in some **binding process** called *dyein-microtubule interaction*. The *dyein-microtubule interaction* has as a participant a **Dynein ATPase** called *dynein*.” [13] In this statement, the interaction between the **Microtubule** called *a pair of doublet microtubules* and the **Dynein ATPase** called *dynein* is inferred from the rule that “if **Object A** and **Object B** participate in the **Process P**, then **Object A** interacts with **Object B**.” The semantic graph for this paper is shown in Fig. 1.
4. “*Flagellar* is a **Flagellum** that has as a part some **Cytoplasmic Structure** called *axoneme*. There is a **Microtubule** that is part of the *axoneme*. There is a **molecular process** that has as an actor the **Microtubule** and that is regulated by some **molecule part** called *dynein arm*.” [21] The match between this statement and the relationship association is obtained basically the same way as with statement 2, with the additional inference from the rule that “if **Object A** is an actor of **Process P** and **Object A** is part of **Object B**, then **Object B** is also an actor of **Process P**.”
5. “There is a **Flagellum** that has as a part some **Cytoplasmic Structure** called *axoneme*. *Glass substrate* is a **physical object** that binds to the *axoneme*.” [22] This statement is a direct match to the relationship association using the subsumption relationship between “binds to” and “interacts with”.

## 5 Related Work

One of the aims of the relationship association extraction technique presented in this paper is to generate new hypotheses from the literature. Several other researchers have presented work with similar aims, as we mentioned in Section 1.

Swanson presented one of the first literature-based hypotheses that fish oil may have beneficial effects in patients with Raynaud’s disease [23]. His original discoveries were based on an exhaustive reading of the literature [24], [25], [26]. Swanson described the process of his literature-based hypotheses discovery with his ABC model, which states that if A and B are related, and B and C are related, then A and C might be indirectly related.

The Arrowsmith system was developed from text analysis scripts based on Swanson’s initial work [5]. The Arrowsmith system examines concepts occurring in the titles of papers from MEDLINE. If two concepts co-occur in a title, then they are considered to be related.

Gordon and Lindsay developed a methodology for replicating Swanson’s discovery based on lexical statistics. In addition to the title words used by the Arrowsmith system, they computed frequency-based statistics of words and multiword phrases from the full MEDLINE records [27], [28].

Weeber and colleagues used the Unified Medical Language System (UMLS) Metathesaurus to identify biomedically interesting concepts in MEDLINE titles and abstracts. They also exploited the semantic categorisation that is included in the UMLS framework [6], [29].

Hristovski and colleagues used the manually assigned MeSH terms rather than the natural language text from MEDLINE citations. Their tool BITOLA computes association rules that measure the relationship between MeSH terms in the form  $X \rightarrow Y$  (*confidence, support*). Although they used term co-occurrence as an indication of a relationship between concepts represented by MeSH terms, they did not try to identify the kind of relationship. Furthermore, their association rules are between two concepts, not two relationships [7], [30].

All these existing approaches focus on extracting non-specified relationships between pairs of concepts in the target domain. In contrast, our approach tries to discover an implicit association between a pair of explicit relationships, each of which predicates the specific way in which a single shared concept is modified by some other concept. We call a pair of relationships that are found to be associated in this way a “relationship association”. Our approach uses several semantic web techniques to enable the extraction of relationship associations. Hristovski *et al.* suggested that MeSH terms may represent more precisely what a particular paper is about than terms extracted from plain text. While we basically agree with their hypothesis, MeSH terms alone cannot represent the relationships between the entities that are described in a paper. Our approach uses classes and properties specified in an ontology, which logically structures a set of MeSH terms, to represent the relationships between entities described in a MEDLINE paper. We believe that these semantic statements, which express the relationships between concepts, are able to provide even more precise representations of that paper.

## 6 Conclusions and Future Work

How to help researchers make scientific discoveries using the existing published literature is an important problem in knowledge discovery, and many literature-based discovery methods and tools have been proposed for solving this problem. However, these approaches mainly use text-mining techniques to discover non-specified relationships between pairs of concepts.

Here, we use semantic web techniques to discover the association of pairs of specified relationships, which we call relationship associations, from semantic graphs describing scientific papers in a computer-interpretable way. These relationship associations could help researchers generate scientific hypotheses and also assist in the creation of semantic graphs for other papers in the same knowledge domain.

We first reviewed our previous work for creating semantic graphs using an ontology developed from a subset of the MeSH vocabulary. Then, we described the process of extracting relationship associations from those semantic graphs. First, we generate triple queries from the semantic graphs and calculate their frequencies of occurrence by matching them with the set of semantic graphs using logical and rule-based inference. Next, we generate association queries from the triple queries that occur in at least two semantic graphs. We remove association queries that specify the same semantic relationships and match the remaining association queries with the set of semantic graphs to get their frequencies of occurrence. Finally, we convert the association queries selected using relevance criteria to relationship associations expressed in natural language.

We applied the approach to a set of 392 semantic graphs based on papers from MEDLINE. The relationship associations that were created from these semantic graphs were examined and several interesting ones were identified.

The relationship associations extracted here can be compared with co-occurrence of terms A and B in the Swanson ABC model, where instead of single concepts we use relationships given by semantic triples. The next step will be to discover indirect associations between two relationships via a shared intermediary relationship, associations that are not explicitly stated in any of the semantic graphs from which the relationship associations were extracted. We are investigating techniques for establishing these implied relationship associations. In addition, we are applying the relationship association extraction method presented here a set of semantic graphs that have been created to express failure events in the field of engineering [31].

**Acknowledgements.** The authors thank Daisuke Hoshiyama for advice on interpretation of the experimentally extracted relationship associations. Funding support for this work was provided by the President's Office of the University of Tokyo.

## References

1. Marrs, K.A., Novak, G.: Just-in-Time Teaching in Biology: Creating an Active Learner Classroom Using the Internet. *Cell Biology Education* 3, 49–61 (2004)
2. King, T.J., Roberts, M.B.V.: *Biology: A Functional Approach*. Thomas Nelson and Sons (1986) ISBN 978-0174480358

3. Langley, P.: The Computational Support of Scientific Discovery. *International Journal of Human-Computer Studies* 53, 393–410 (2000)
4. Racunas, S.A., Shah, N.H., Albert, I., Fedoroff, N.V.: HyBrow: a Prototype System for Computer-Aided Hypothesis Evaluation. *Bioinformatics* 20 (suppl. 1), i257–i264 (2004)
5. Swanson, D.R., Smalheiser, N.R.: An Interactive System for Finding Complementary Literatures: a Stimulus to Scientific Discovery. *Artificial Intelligence* 91, 183–203 (1997)
6. Weeber, M., Kors, J.A., Mons, B.: Online Tools to Support Literature-Based Discovery in the Life Sciences. *Briefings in Bioinformatics* 6(3), 277–286 (2005)
7. Hristovski, D., Peterlin, B., Mitchell, J.A., Humphrey, S.M.: Using Literature-Based Discovery to Identify Disease Candidate Genes. *International Journal of Medical Informatics* 74(2-4), 289–298 (2005)
8. Srinivasan, P.: Text Mining: Generating Hypotheses from MEDLINE. *JASIST* 55(5), 396–413 (2004)
9. Hallett, C., Scott, D., Power, R.: Composing Questions through Conceptual Authoring. *Computational Linguistics* 33(1), 105–133 (2007)
10. Natarajan, J., Berrar, D., Hack, C.J., Dubitzky, W.: Knowledge Discovery in Biology and Biotechnology Texts: A Review of Techniques, Evaluation Strategies, and Applications. *Critical Reviews in Biotechnology* 25(1), 31–52 (2005)
11. Kraines, S., Guo, W., Kemper, B., Nakamura, Y.: EKOSS: A Knowledge-User Centered Approach to Knowledge Sharing, Discovery, and Integration on the Semantic Web. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 833–846. Springer, Heidelberg (2006)
12. Kraines, S., Makino, T., Mizutani, H., Okuda, Y., Shidahara, Y., Takagi, T.: Transforming MeSH into DL for Creating Computer-Understandable Knowledge Statements (in preparation)
13. Aoyama, S., Kamiya, R.: Cyclical Interactions between Two Outer Doublet Microtubules in Split Flagellar Axonemes. *Biophys J.* 89(5), 3261–3268 (2005)
14. Cohen, W.W., Ravikumar, P., Fienberg, S.E.: A Comparison of String Distance Metrics for Name-Matching Tasks. In: Proceedings of the ACM Workshop on Data Cleaning, Record Linkage and Object Identification, Washington, DC (August 2003)
15. Racer Systems GmbH & Co. KG, <http://www.racer-systems.com>
16. Guo, W., Kraines, S.: Explicit Scientific Knowledge Comparison Based on Semantic Description Matching. In: American Society for Information Science and Technology 2008 Annual Meeting, Columbus, Ohio (2008)
17. OWL Web Ontology Language Guide, <http://www.w3.org/TR/owl-guide/>
18. Kraines, B., Guo, W.: Using Human Authored Description Logics ABoxes as Concept Models for Natural Language Generation. In: American Society for Information Science and Technology 2009 Annual Meeting, Vancouver, British Columbia, Canada (2009)
19. Morita, Y., Shingyoji, C.: Effects of Imposed Bending on Microtubule Sliding in Sperm Flagella. *Current Biology* 14(23), 2113–2118 (2004)
20. Nakano, I., Kobayashi, T., Yoshimura, M., Shingyoji, C.: Central-Pair-Linked Regulation of Microtubule Sliding by Calcium in Flagellar Axonemes. *Journal of Cell Science* 116(8), 1627–1636 (2003)
21. Yanagisawa, H., Kamiya, R.: A Tektin Homologues is Decreased in Chlamydomonas Mutants Lacking an Axonemal Inner-Arm Dynein. *Molecular Biology of the Cell* 15(5), 2105–2115 (2004)
22. Sakakibara, H.M., Kunioka, Y., Yamada, T., Kamimura, S.: Diameter Oscillation of Axonemes in Sea-Urchin Sperm Flagella. *Biophys J.* 86(1 Pt 1), 346–352 (2004)

23. Swanson, D.R.: Fish oil, Raynaud's Syndrome, and Undiscovered Public Knowledge. *Perspectives in Biology and Medicine* 30, 7–18 (1986)
24. Swanson, D.R.: Migraine and Magnesium: Eleven Neglected Connections. *Perspectives in Biology and Medicine* 31, 526–557 (1988)
25. Swanson, D.R.: Somatomedin C and Arginine: Implicit Connections between Mutually Isolated Literatures. *Perspectives in Biology and Medicine* 33(2), 157–179 (1990)
26. Swanson, D.R., Smalheiser, N.R., Bookstein, A.: Information Discovery from Complementary Literatures: Categorizing Viruses as Potential Weapons. *JASIST* 52(10), 797–812 (2001)
27. Gordon, M.D., Lindsay, R.K.: Toward Discovery Support Systems: A Replication, Re-Examination, and Extension of Swanson's Work on Literature-Based Discovery of a Connection between Raynaud's and Fish Oil. *JASIST* 47(2), 116–128 (1996)
28. Lindsay, R.K., Gordon, M.D.: Literature-Based Discovery by Lexical Statistics. *JASIST* 50(7), 574–587 (1999)
29. Weeber, M., Vos, R., Klein, H., de Jong-van den Berg, L.T.W., Aronson, A.R., Molema, G.: Generating Hypotheses by Discovering Implicit Associations in the Literature: A Case Report of a Search for New Potential Therapeutic Uses for Thalidomide. *J. American Medical Informatics Association* 10(3), 252–259 (2003)
30. Hristovski, D., Stare, J., Peterlin, B., Dzeroski, S.: Supporting Discovery in Medicine by Association Rule Mining in Medline and UMLS. *Medinfo* 10(Pt2), 1344–1348 (2001)
31. Guo, W., Kraines, S.B.: Mining relationship associations from knowledge about failures using ontology and inference. In: Perner, P. (ed.) *ICDM 2010. LNCS*, vol. 6171, pp. 617–631. Springer, Heidelberg (2010)

# Sequential Supervised Learning for Hypernym Discovery from Wikipedia

Berenike Litz, Hagen Langer, and Rainer Malaka

TZI, University of Bremen, Bremen, Germany

berenike.litz@gmail.com, {hlanger,malaka}@tzi.de

**Abstract.** Hypernym discovery is an essential task for building and extending ontologies automatically. In comparison to the whole Web as a source for information extraction, online encyclopedias provide far more structuredness and reliability. In this paper we propose a novel approach that combines syntactic and lexical-semantic information to identify hypernymic relationships. We compiled semi-automatically and manually created training data and a gold standard for evaluation with the first sentences from the German version of Wikipedia. We trained a sequential supervised learner with a semantically enhanced tagset. The experiments showed that the cleanliness of the data is far more important than the amount of the same. Furthermore, it was shown that bootstrapping is a viable approach to ameliorate the results. Our approach outperformed the competitive lexico-syntactic patterns by 7% leading to an  $F_1$ -measure of over .91.

**Keywords:** Information extraction, Hypernym discovery, Sequential supervised learning, Hidden Markov models, Syntactic-semantic tagging.

## 1 Introduction

One of the main groups of named entities from the Message Understanding Conferences (MUC) is the group of *persons*. Finding the direct hypernym of a person is an important subtask of fine-grained named entity classification and of ontology extension and is, therefore, subject to this research. Due to their structuredness and reliability, encyclopedias present a highly important source for all types of information extraction. Moreover, in the case of the easily extendable online encyclopedia Wikipedia, a very good coverage of entries can be expected. It can be observed that there is a repetitive structure in encyclopedia texts such as “X is/was a Y”. These patterns are also found in Wikipedia entries for persons as shown in the following examples:

- (1) **Ferdinand Magellan** was a Portuguese **explorer**.
- (2) **Michael Ballack** is a German **football midfielder** who plays for Chelsea of the English Premier League and is the captain of the German national team.

The most common method for hypernym discovery is the deployment of lexico-syntactic patterns [1,2,3,4]. One shortcoming of all the named approaches is that they are applied to a data source, in which the recall of the patterns is naturally low. The

state-of-the-art method [5] reports an  $F_1$ -value in the lower third. To our knowledge, only [6] and [7] have applied Wikipedia as a data source for lexico-syntactic patterns and both achieved results of nearly .88  $F_1$ . As the patterns have proven to be simple to implement and unequally powerful on encyclopedia resources, they were applied as a baseline approach.

When observing the syntactic structure of the first sentences of Wikipedia entries, it becomes obvious that syntax is the key to discovering hypernyms. The only word tagged as a Noun in Example (3) is also the hypernym of the Proper Noun Ferdinand\_Magellan in this context.

- (3) Ferdinand\_Magellan/Proper-Noun was/Finite-Verb a/Article Portuguese/Adjective explorer/Noun.

Apart from lexico-syntactic patterns, all nouns of the first sentences could, therefore, be considered as hypernyms for a second heuristic baseline approach.

However, heuristic methods have shortcomings. The simple employment of syntactic information is comparably low in precision whereas the lexico-syntactic patterns might be rather low in recall. In contrast to this, a sequential supervised learning approach which combines syntactic and lexical-semantic information to identify hypernymic relationships might prove to outperform these straightforward and state-of-the-art methods.

For this purpose, a classical part-of-speech tagger can be applied with an extended tagset in order to create a syntactic-semantic tagger. When some of the syntactic tags are replaced by lexical-semantic tags, the tagger might not only predict the part of speeches of words but also their lexical-semantic relationships. Example (4) shows, how the tagset needs to be adjusted for the task of finding the Hypernym of a certain Person.

- (4) Ferdinand\_Magellan/**Person** was/Finite-Verb a/Article Portuguese/Adjective explorer/**Hypernym**.

The research presented herein was conducted for the German language. There exist a number of part-of-speech taggers for German that can be applied and adjusted for the approach. The only prerequisite is that the tagger can be trained with a new trainset, which is true for most statistical taggers. Further, it is of advantage that the taggers provide a German model, which can be utilized for preprocessing the new data.

The statistical taggers qtag [8], Trigrams'n'Tags [9], TreeTagger [10] and Stanford Log-linear Part-Of-Speech Tagger [11] all provide a German model and perform with an average accuracy of around 97%. Positive experience with Trigrams'n'Tags (TnT) on the task of semantic tagging [12] and a comparison of TnT with qtag, however, resulted in choosing TnT for the task.

To sum up, the task of this work is to discover hypernyms of persons from Wikipedia articles. Two baseline approaches and a syntactic-semantic tagger were implemented. The first one simply considers every word identified as a noun by a part of speech tagger as a hypernym. The second baseline applies lexico-syntactic patterns in order to find hypernyms. The syntactic-semantic tagger identifies hypernyms by taking probabilistic methods and statistical information about syntactic context and information about the

structural distribution of words denoting persons and their corresponding hypernyms into account.

In the following, the data collection and preparation is described that is necessary for training the syntactic-semantic tagger and for comparing the results of the baseline approaches and the tagger.

## 2 Data Collection and Preparation

For a comparison of all approaches, a gold standard was created. For the syntactic-semantic tagger, furthermore, training data had to be assembled, which consist of the first paragraphs of Wikipedia articles with tags giving syntactic and lexical-semantic information.

*Training Data.* For training the syntactic-semantic tagger, a pre-annotated data set is needed. The first paragraphs of the Wikipedia entries were, therefore, annotated with the help of the part-of-speech (POS) tagger TnT and with the tags Person (in case the token was the name of the person the article was about) and Hypernym (in case the token was a hypernym of Person), which replaced the POS tags. Example (5) shows the annotation.

- (5) Ferdinand\_Magellan/Person war/Finite-Verb ein/Article portugiesischer/Adjective Seefahrer/Hypernym  
 (In English: Ferdinand\_Magellan/Person was/Finite-Verb a/Article Portuguese/Adjective navigator/Hypernym)

Wikipedia articles include metadata for different named entity types, which can be applied for such an annotation. For the purpose of finding hypernyms of persons the so-called *person data* are valuable that are included in articles about persons to be automatically extracted and processed<sup>1</sup>. The data consists of fields such as name, birthdate and -place, deathdate and -place and a short description of the person. For the task of hypernym discovery, the fields of name and short description were applied. A database was created, which includes these fields as well as the texts of the Wikipedia entries. For this work, all 70,000 entries in the database were used.

Instead of manually annotating the whole data, the short description was used to extract hypernyms automatically. As nouns are generally capitalized in German, all capitalized words were considered to be nouns and, therefore, hypernyms. The short description in Example 6 for the Portuguese sailor Magellan could be utilized to extract *Seefahrer* (*sailor*) and *Krone* (*crown*).

- (6) portugiesischer **Seefahrer**, der für die spanische **Krone** segelte (In English: Portuguese **sailor** who sailed for the Spanish **crown**)

POS tagging was not applied for the short descriptions, as those often only include sentence fragments or single words and a tagger could not yield reliable results. However, in German, the selection of all capitalized words as nouns and named entities is

<sup>1</sup> For German: <http://de.wikipedia.org/wiki/Hilfe:Personendaten> (last access: February 8 2010);  
 For English: <http://en.wikipedia.org/wiki/Wikipedia:Persondata> (last access: February 8 2010).

nearly unfailing. The wrong annotation of non-hypernyms as e.g. *crown* in this example was tolerated in consideration of getting hold of a large amount of completely automatically annotated training data.

Different training sets were created in order to find the set which leads to the best results. In all cases the untagged texts were the first paragraphs of Wikipedia articles, however, the tagging was done by differing means and the size of training data varied.

First, a training set was created with 300,000 tokens that was tagged by the POS tagger. The tag Person was assigned to the named entities, which were the subject of the article. Each noun of the short description was tagged as Hypernym. As the information of the person data was obtained without additional manual labor, this training set is referred to as the *semi-automatically created trainset* in the following.

The semi-automatic creation of training data is error prone to some extent as was shown with Example 6. Therefore, parts of the existing training set were corrected. Part of the semi-automatically created training data was proof-read by a person and in all wrong cases the Hypernym tag was replaced or inserted. In the beginning, about 1% of the whole training data were corrected, then 2%, 3% and 4%. These parts were utilized as completely *manually created training sets*.

*Gold Standard.* For the creation of the gold standard, a separate set from the training data with 4,000 tokens was corrected by an annotator and rechecked by another person. This data not only included the first sentences but also larger extracts from the Wikipedia texts, so that the gain and loss in precision according to the various approaches are apparent. The untagged version of the test set was then tagged by each of the proposed methods and then compared with the gold standard. All together, the set contained 450 hypernyms.

The following sections present the two baseline approaches and the syntactic-semantic tagger.

### 3 Baselines: Syntactic Information and Lexico-Syntactic Patterns

The first paragraphs of encyclopedia entries usually contain a hypernym of the word in focus as well as few other nouns. Hence, it makes sense to take syntactic information about the words into account. The simplest approach in this case is to tag any token marked as a noun (NN in the tagset) by a POS tagger as a Hypernym. It can be expected that the recall of this approach is very high with a loss in precision.

A more sophisticated approach is the employment of lexico-syntactic patterns. In contrast to the previous method, this one is expected to be high in precision with a loss in recall. Analyzing the data, it appeared that the Patterns (4) and (5), derived with the rules in (1), (2) and (3), are representable for most examples. For both patterns the typical constituents of a noun phrase (NP) are important (see (1)), which are article (ART), adjective (ADJA), noun (NN) and named entity (NE). Even though a hypernym is generally not a named entity, this tag was included in the pattern as the tagger classifies unseen nouns often wrongly as named entities. The star (“\*”) behind a bracket marks a possible number of occurrence between zero and more times for the expression in the bracket. The pipe (“|”) denotes alternatives in this expression, which means to find the left hand OR right hand values, in this case, NN or NE. The German verb forms *war*,

*ist*, *waren* and *sind* are inflected forms of the verb *sein* (in English: *to be* with inflected forms *was*, *is*, *were*, *are*) and are symbolized by VAFIN. The conjunction *und* (English: *and*) is symbolized by KON.

$$NP = (ART)^* (ADJA)^* (NN|NE) \quad (1)$$

$$VAFIN = (\text{war}|\text{ist}|\text{waren}|\text{sind}) \quad (2)$$

$$KON = \text{und} \quad (3)$$

Pattern (4) takes the tag Person into account, which means that the topic of the Wikipedia entry is part of the expression. The hypernym would be any noun (NN) or named entity (NE) found in this pattern.

$$(\text{Person}) (VAFIN) (NP,)^* (NP) ((KON) (NP))^* \quad (4)$$

The data of Wikipedia texts is only structured to some extent as it is encoded mostly by volunteers. Therefore, a pattern disregarding the Person tag should also be tested, as the recall is likely to be higher (see Pattern (5)).

$$(VAFIN) (NP,)^* (NP) ((KON) (NP))^* \quad (5)$$

## 4 Syntactic-Semantic Tagging

A supervised tagger needs to be trained with pre-annotated data, in order to be able to predict the tags of unseen data. During the training phase, the lexicon and an n-gram list are created by TnT. These lists contain the frequencies of tokens and tags in the training data and are needed for the prediction of the probability for a specific tag at a specific location. The lexicon contains for each token the frequency of its tag as it occurred in the training data. These frequencies are needed to determine lexical probabilities in the tagging process. The n-gram list contains the contextual frequencies for uni-, bi- and trigrams. With the help of the lexicon and the n-gram list, predictions about the probability of a particular tag can be made. E.g. *Mechaniker* (Engl.: mechanic) was tagged as NN and as Hypernym. As the n-gram list indicates that after VAFIN the occurrence of the tag Hypernym is much more likely, *Mechaniker* will be tagged as Hypernym in this context. The n-gram list and the lexicon are applied by the tagger to predict the most likely tag sequence for unseen word sequences.

Apart from the default settings of the tagger, it is possible to choose options concerning the suffix length and the values for linear interpolation. Unknown words in TnT are handled by taking into account the word ending, which is also important for hypernyms of persons, as those often refer to professions. For instance, a frequent ending for nouns referring to professions of persons in German is *-er* (e.g. *Maler*, *Musiker*, *Politiker*; In English: painter, musician, politician) and the corresponding female word form of it with the ending *-erin* (correspondingly, *Malerin*, *Musikerin*, *Politikerin*). TnT provides an option for changing the suffix length in order to find the best setting for a particular training set that was utilized for finding the best suffix length for the data. TnT utilizes a method developed by Samuelsson [13].

Due to the sparse-data problem, trigram probabilities generated from a corpus cannot be directly used in most cases. The default setting for TnT is computed with *deleted interpolation*. This is an individual adjustment of the values according to the model. The algorithm successively removes each trigram from the training corpus and estimates best values for the  $\lambda$ s from all other n-grams in the corpus. However, a manual adjustment of the  $\lambda$  values might improve the results of the trained model.

Apart from finding the best values of the named options, experiments with the training data size of the different training sets were conducted. Further, an approach was applied which, on the one hand, makes use of the manually created training data, and, on the other hand, can countervail cost-intensive annotation. This kind of approach is referred to as *bootstrapping* [14]. Bootstrapping incorporates the notion of "the plenitude of unlabeled natural language data, and the paucity of labeled data" [14]. For the experiments, the model created by the manually annotated training data was utilized and the remaining part of the unlabeled data was split into smaller portions of text. In particular, the model that was trained with 4% of the overall training data and the unlabeled version of the remaining 96% of the training data were utilized.

The bootstrapping is conducted in the following five steps:

1. Split unlabeled data into files containing 5,000 tokens
2. Take manually created train set to tag first unlabeled data file
3. Concatenate the manually labeled data and the newly tagged file
4. Create a new model with the concatenated file
5. Continue with Step 2 until all split files are tagged

To sum up, the syntactic-semantic tagger was trained with a number of training sets. Options for suffix length and linear interpolation were investigated and a bootstrapping approach was conducted. The evaluation will include results for all named options.

## 5 Evaluation and Results

The quality of the results was measured by the values *precision* and *recall* and their harmonic mean, also known as the *F<sub>1</sub>-measure* [15]. Apart from that *accuracy* is quoted in particular cases, which is the standard measure for tagging tasks.

### 5.1 Baseline 1: Syntactic Information

For the POS tagging, two Second Order Hidden Markov Model taggers were compared. The first one was the Trigrams' n'Tags (TnT) Tagger [9] and the second one qtag [8]. Both taggers were trained with the POS tagged version of the NEGRA 2 corpus<sup>2</sup>. With an *F<sub>1</sub>* of .55 the results of TnT were much more promising than the ones of qtag with .47 for the approach taking only nouns (NNs) into account. These results led to the

---

<sup>2</sup> The NEGRA corpus version 2 consists of 355,096 tokens (20,602 sentences) of German newspaper text, taken from the Frankfurter Rundschau. For more information visit: <http://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/negra-corpus.html> (last access: February 8 2010).

decision to apply TnT not only for this baseline but also for training the syntactic-semantic tagger.

As in many cases nouns (NN) were tagged by the POS tagger as named entities (NE), the recall of the best method was only 79%. Once the named entities are replaced by the Hypernym tag as well, the recall yielded 97% with a further loss in precision. The loss in precision gave reason to the resulting lower  $F_1$  of the approach including named entities. Therefore, the approach taking only nouns into account outperformed this one by .03  $F_1$  for qtag as well as for TnT.

## 5.2 Baseline 2: Lexico-Syntactic Patterns

The lexico-syntactic patterns presented in Equations (4) and (5) were applied to the test data and the results were compared with the gold standard. Pattern (4) had the constraint that the person name is included in the sequence with a hypernym. Pattern (5) did not have this constraint. Both patterns were tested with and without taking NEs into account as hypernyms. The precision of Pattern (4) which only takes NNs into account, was highest. Still, the precision values of all patterns were comparable. Highest overall results with an  $F_1$  value of .85 were achieved for Pattern (5) excluding person names and including NEs. These values were utilized for a comparison with the performance of the syntactic-semantic tagger.

## 5.3 Syntactic-Semantic Tagging

The evaluation of the syntactic-semantic tagging approaches is presented in the following. As a first step, the performance of the semi-automatically created model was evaluated with respect to the amount of training data. Afterwards, these results were compared with a small amount of completely manually annotated training data. With the help of the best training set from these evaluations, the best suffix length for the suffix tries for unknown words was calculated as well as the best results for linear interpolation. The results from the suffix length and the results of linear interpolation were, then, taken into account for a bootstrapping method. The bootstrapping approach was evaluated with respect to the best choice of parameter adjustments and the number of included files for the training data.

*Semi-automatically Created Training Data.* First experiments with the size of the semi-automatically created training data showed that the total size of around 300,000 tokens played only an inferior role for the quality of the outcome. It turned out that even 1.25% of the data yielded promising results with an  $F_1$ -value of .73. 12.5% of the data yielded .76  $F_1$  and all 100% only slightly better results with .78  $F_1$ .

The conclusion of these findings is that even considerably smaller data sets can be taken into account for this approach. This is particularly interesting for approaches, where no pre-annotated data is available. Even though the number of unknown tokens increases with a decrease in training data, the tagger performs competitively.

*Manually Annotated Training Data.* The findings of the data size described in the previous paragraph led to the idea of manually correcting a small amount of the semi-automatically created training data. First, only about 1% of the whole data were corrected, then 2%, 3% and finally 4% to see the improvement on the results.

The results showed that the quality of results increases slightly but steadily with the size of manually annotated data. On 1% of the data, the  $F_1$ -value of .85 already outperformed the results of the semi-automatically created model. For 4% of the data, the improvement was .03 leading to an  $F_1$ -value of .88.

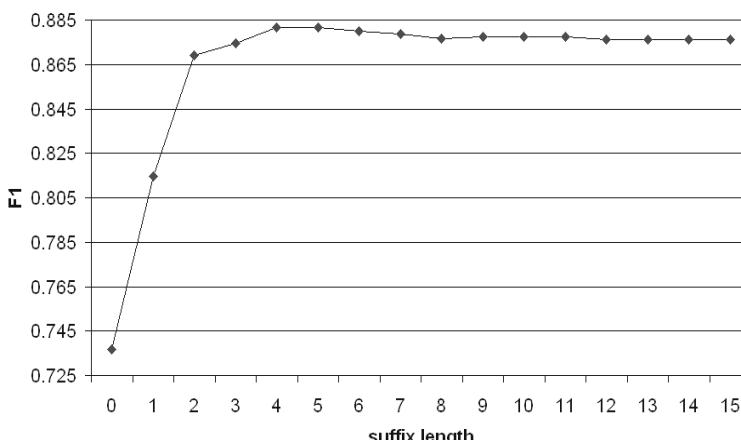
As the tagger trained with a corrected version of 4% of the total training data outperformed the other ones, it was taken for further experiments.

*Suffix Length.* For 1.25% of the training data, the percentage of unknown tokens was 42% and for 100% it was only 19%. However, the higher percentage of unknown tokens for smaller data sets only slightly decreased the accuracy in the training data. There, the accuracy value varied between 0.936 and 0.948 for the corresponding percentages.

The word ending is an indication of the presumptive part-of-speech of an unknown word in a corpus. For the task of hypernym discovery this is also applicable. Experiments were conducted with varying suffix length in order to examine if the TnT default length=10 was the best choice for the task.

The results for precision of suffix length=0 outperformed the other ones considerably. However, recall underperformed at this suffix length. This phenomenon can be explained by the following fact: The technique of taking a suffix into account is utilized in order to provide the robustness of natural language processing approaches. Recall is an important issue here and can be boosted with a suffix analysis. Nonetheless, a trade off with precision takes place. Even though, the high precision result appeared promising for improvements, later evaluations showed that recall could not be increased with a remaining precision value.

The results in Figure 1 show that the suffix lengths of 4, 5, 6 and 7 were very similar for  $F_1$ , and outperformed the default of 10. The evaluation of the best option settings for the bootstrapping approach will, therefore, consider this fact.

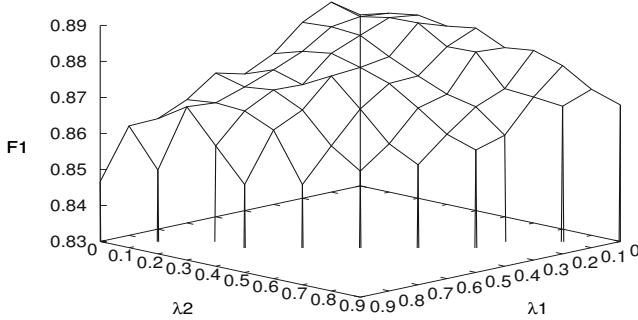


**Fig. 1.** Results of the evaluation of suffix length for 4% of manually annotated training data for  $F_1$ . Suffix length 0 to 15

*Linear Interpolation.* TnT applies linear interpolation for smoothing of trigram probabilities:

$$P(y_3|y_1, y_2) = \lambda_1 \hat{P}(y_3) + \lambda_2 \hat{P}(y_3|y_2) + \lambda_3 \hat{P}(y_3|y_1, y_2) \quad (6)$$

$\lambda_1$  and  $\lambda_2$  determine the extent of smoothing applied on the data. In case  $\lambda_3$  is set to 1, no smoothing takes place. That means, the lower the values for  $\lambda_1$  and  $\lambda_2$ , the lower is the extent of smoothing.



**Fig. 2.**  $F_1$  results for all possible  $\lambda_1$  and  $\lambda_2$  values for the linear interpolation of the syntactic-semantic tagger

The default setting for linear interpolation of TnT is computed with deleted interpolation. In the case of the model, the values  $\lambda_1 = 0.163$ ,  $\lambda_2 = 0.246$  and  $\lambda_3 = 0.591$  were calculated by this algorithm. To further improve the results of the approach, these values were modified for experiments. Therefore, the combination of all possible values for  $\lambda_1$ ,  $\lambda_2$  and  $\lambda_3$  was evaluated in the range of [0,1] with an interval size of 0.1. The sum of  $\lambda_1$ ,  $\lambda_2$  and  $\lambda_3$  is always 1:

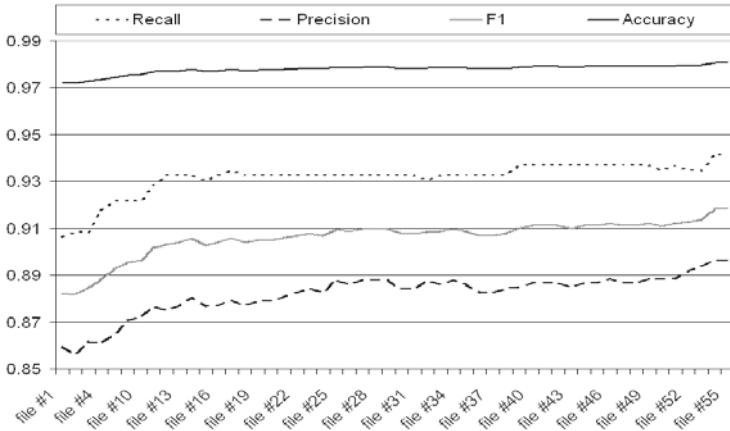
$$\lambda_3 = 1 - \lambda_1 - \lambda_2 \quad (7)$$

In the following figures, only  $\lambda_1$  and  $\lambda_2$  are shown, which is sufficient as  $\lambda_3$  is implicitly defined by  $\lambda_1$  and  $\lambda_2$ .

Figure 2 shows the  $F_1$  results for the choices with the dimensions  $\lambda_1$  and  $\lambda_2$ . An adjustment to  $\lambda_1 = 0.1$   $\lambda_2 = 0.0$  yields the best results. The  $F_1$ -values were improved from .8808 to .8827 in comparison with the results of the tagger applying deleted interpolation. Even if the improvement of manually adjusting the  $\lambda$ -values was marginal, for a bootstrapping approach a considerable improvement is expected to be achieved.

The precision values showed that precision is clearly higher in cases with low values for  $\lambda_1$  and  $\lambda_2$ . Whereas, the recall values showed that a very low value for  $\lambda_3$  and values for  $\lambda_1$  and  $\lambda_2$  in the range [0.3,0.5] are optimal. This event can be explained by the fact, that linear interpolation is generally applied to improve recall and that less smoothing is advantageous for precision.

*Bootstrapping of Training Data.* The increasing  $F_1$  values for an increasing amount of manually annotated training data gave reason to conduct experiments compensating



**Fig. 3.** Bootstrapping for the syntactic-semantic tagger including linear interpolation with  $\lambda_1 = 0.1$  and  $\lambda_2 = 0.0$  and suffix length=6

lacking training data. However, the first experiments with the semi-automatically created data were not successful. Therefore, a bootstrapping approach was implemented.

First results were already as high as  $F_1 = .904$ . The findings of the best choice of suffix length and for the  $\lambda$ s for linear interpolation were then utilized to boost these results. As one individual value for the best suffix length was not apparent, each of the values 4 to 7 was evaluated. It appeared that the choice of suffix length=6 yielded highest results. Figure 3 includes the results of the best performing model with an  $F_1$  value of 0.9181722.

**Comparison of Approaches.** The comparison of the results in Table 1 yielded by the best performing syntactic-semantic tagger and the syntactic information and lexico-syntactic pattern approaches showed that the syntactic information baseline outperformed the lexico-syntactic patterns slightly with the recall value of 0.771 but was inferior in all other aspects. The highest recall was gained by the statistical approach with over 0.937. Precision was very low for the syntactic information baseline with under 0.432. The lexico-syntactic patterns outperformed the syntactic-semantic tagger here by 0.07 with 0.969. As recall was much higher for the syntactic-semantic tagger, it outperformed the patterns with results of as high as .918  $F_1$ .

**Table 1.** Comparison of the results of the two baseline approaches and the syntactic-semantic tagger

	Recall	Prec.	F1
Syntactic Information	0.771	0.431	0.553
Lexico-Syntactic Patterns	0.758	<b>0.969</b>	0.851
Syntactic-Semantic Tagger	<b>0.937</b>	0.9	<b>0.918</b>

A clear advantage of the syntactic-semantic tagger is its access to the lexicon. In a number of cases it could be observed, that the tagger did tag hypernyms correctly even though the context did not indicate it. Examples are “Darsteller”, “Pfarrer” or “Prinzessin” (Engl.: “performer”, “clergyman” or “princess”). The combination of the lexicon and a suffix analysis helped the tagger in finding hypernyms which were not part of the lexicon such as e.g. “Wanderprediger”, “Esoterikasammler” or “Reichswirtschaftsminister” (Engl.: “itinerant preacher”, “esoterics collector” or “Reich economics minister”). However, even though these features generally helped in finding hypernyms, in some cases also the wrong words were tagged as hypernyms. E.g. the word “Heiliger” can denote the noun “saint” but also the adjective. In the context “Heiliger Capestranus”, it is an adjective but capitalized as an addition to a named entity. As the lexicon lists “Heiliger” most often as a hypernym it was wrongly tagged as such. Further, the suffix analysis fails when it comes to nouns with untypical suffixes as e.g. “Dienstmagd” (Engl.: “maidservant”).

These findings are an important issue for future work on this topic as discussed in the conclusions section.

## 5.4 Conclusions

In this paper, approaches for hypernym discovery from encyclopedias were presented and compared to each other. One approach took the syntactic information into account to predict a hypernym, which resulted in a very high recall and low precision. Another approach applied lexico-syntactic patterns that reflected the most likely word order in which a hypernym follows a hyponym in the given data set. The evaluation showed very good results with respect to precision and also recall was high. As the lexico-syntactic patterns performed very well on the data, they are likely to be used as a cheap and robust straightforward solution for similar data.

Apart from that, a syntactic-semantic tagger was presented. Even though the lexico-syntactic patterns performed competitively with the syntactic-semantic tagger in the first test, with an optimization of the model the tagger could outperform them. The first tests utilized a semi-automatically created training data set containing 300,000 tokens, which resulted in  $.78 F_1$ . It appeared that a fractional amount of the data (around 4%) that was manually corrected, outperformed those results with an  $F_1$  of .88. A method for detecting unknown tokens with evidence from the suffix trie of length 6 and an adjustment of the  $\lambda$  values for linear interpolation only gave slight improvements on the third position of the decimal point. However, for the bootstrapping approach these minimal improvements became apparent and the best model obtained with bootstrapping resulted in an outstanding  $F_1$ -value of over .91. Admittedly, an optimization of the training model on only one gold standard leads to biased results and the applicability of the model to new domains will not be possible without further ado. However, the outstanding results of the tagger with the bootstrapped model show the great potential of the method. The presented results of the lexico-syntactic patterns and the syntactic-semantic tagger make clear that hypernym discovery from encyclopedias is a powerful way of information extraction and are both groundbreaking for extracting entities linked through other semantic relations.

For future work, especially the qualitative analysis of the comparison of approaches is instructional. On the one hand, the lexico-syntactic patterns would benefit from a lexicon such as the one produced by the syntactic-semantic tagger. The syntactic-semantic tagger, on the other hand, would benefit from the clear and symbolic structure such as the one provided by the patterns. In a new use case, such as e.g. the extraction of hypernyms for all kinds of locations, the patterns will outperform the syntactic-semantic tagger as they are not aligned to a domain-dependent lexicon. Therefore, a combination of the patterns and a syntactic-semantic tagger with an adjusted model for the new domain are predicted to prove nearly as powerful as a manual extraction.

Future Work will further involve the deployment of other sequential supervised learning approaches to the given data. One choice is to apply conditional random fields [16]. Those are applied for the task of attribute value extraction [17] and are expected to work well on the task of hypernym discovery. Here, the evaluation of the performance of hidden Markov models and other sequential supervised learning methods on an unseen domain is an important step towards generalization.

**Acknowledgements.** This research was only possible with the financial support of the Klaus Tschira Foundation and the CONTENTUS Use Case of the THESEUS Program funded by the German Federal Ministry of Economics and Technology (BMWi).

## References

1. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the Conference on Computational Linguistics (COLING), Nantes, France (1992)
2. Caraballo, S.A.: Automatic construction of a hypernym-labeled noun hierarchy from text. In: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, Morristown, NJ, USA, Association for Computational Linguistics, pp. 120–126 (1999)
3. Kliegr, T., Chandramouli, K., Nemrava, J., Svatek, V., Izquierdo, E.: Combining image captions and visual analysis for image concept classification. In: Proceedings of the 9th International Workshop on Multimedia Data Mining (MDM), pp. 8–17. ACM, New York (2008)
4. Kozareva, Z., Riloff, E., Hovy, E.: Semantic class learning from the web with hyponym pattern linkage graphs. In: Proceedings of the Annual Meeting on Association for Computational Linguistics (ACL), Columbus, Ohio, Association for Computational Linguistics, pp. 1048–1056 (June 2008)
5. Snow, R., Jurafsky, D., Ng, A.Y.: Semantic taxonomy induction from heterogenous evidence. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Sydney, Australia, Association for Computational Linguistics, pp. 801–808 (July 2006)
6. Kliegr, T., Chandramouli, K., Nemrava, J., Svatek, V., Izquierdo, E.: Wikipedia as the premiere source for targeted hypernym discovery. In: Proceedings of the Wikis, Blogs, Bookmarking Tools - Mining the Web 2.0 Workshop co-located with the 18th European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (2008)
7. Kazama, J., Torisawa, K.: Exploiting wikipedia as external knowledge for named entity recognition. In: Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 698–707 (2007)

8. Tufis, D., Mason, O.: Tagging romanian texts: a case study for qtag, a language independent probabilistic tagger. In: Proceedings of the 1st International Conference of Language Resources and Evaluation (LREC), Granada, Spain (1998)
9. Brants, T.: TnT – A statistical Part-of-Speech tagger. In: Proceedings of the Sixth Conference on Applied Natural Language Processing (ANLP), Seattle, Washington, pp. 224–231 (2000)
10. Schmidt, H.: Probabilistic part-of-speech tagging using decision trees. In: Proceedings of the International Conference on New Methods in Language Processing (NeMLaP), Manchester, U.K., pp. 14–16 (September 1994)
11. Toutanova, K., Manning, C.D.: Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In: Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, Morristown, NJ, USA, Association for Computational Linguistics, pp. 63–70 (2000)
12. Loos, B., Porzel, R.: Resolution of lexical ambiguities in spoken dialogue systems. In: Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue (SIGdial), Morristown, NJ, USA, Association for Computational Linguistics (2004)
13. Samuelsson, C.: Morphological tagging based entirely on bayesian inference. In: Eklund, R. (ed.) Proceedings of the 9th Scandinavian Conference on Computational Linguistics, Stockholm, Sweden, pp. 225–238 (1994)
14. Abney, S.: Bootstrapping. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Morristown, NJ, USA, Association for Computational Linguistics, pp. 360–367 (2002)
15. Van Rijsbergen, C.J.K.: Information Retrieval, 2nd edn., Dept. of Computer Science, University of Glasgow (1979), doi:Van Rijsbergen, C.J.K
16. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of International Conference on Machine Learning (ICML), pp. 282–289 (2001)
17. Loos, B., Biemann, C.: Supporting web-based address extraction with unsupervised tagging. In: Bock, H.H., Gaul, W., Vichi, M. (eds.) Studies in Classification, Data Analysis, and Knowledge Organization. Springer, Heidelberg (2007)

# Evolving Estimators for Software Project Development

Athanasiros Tsakonas and Georgios Dounias

University of the Aegean, Decision and Management Engineering Laboratory  
Kountouriotou 41, 82100 Chios, Greece  
[tsakonas@stt.aegean.gr](mailto:tsakonas@stt.aegean.gr), [g.dounias@aegean.gr](mailto:g.dounias@aegean.gr)

**Abstract.** In this research, an application of a computational intelligence approach for effort estimation in software projects is presented. More specifically, the authors examine a genetic programming system for symbolic regression; the main goal is to derive equations for estimating the development effort that are highly accurate. These mathematical formulas are expected to reveal relationships between the available input features and the estimated project work. The application of the proposed methodology is performed in two software engineering domains. The proposed model is shown capable to produce short and handy formulas that are more precise than the existent in literature.

**Keywords:** Software engineering, Effort estimation, Genetic programming, Symbolic regression.

## 1 Introduction

Nowadays one of the challenges software project managers encounter is the estimation of labor effort for development, as this affects dramatically the project cost. Therefore, in the past, there has been systematic research for models that effectively perform effort estimation. Such models include predictive parametric approaches, such as the COCOMO [1] and the Price S [2]. Other models can also be used for effort estimation, ranging from the so-called historical analogy, mathematical models, to rules-of-thumb. Models that use historical analogy do their assessment using analogies from past projects. Mathematical models, on the other hand, provide relationships between some attributes of the project, usually derived by humans. Nevertheless, the evolving character of software development styles prevented many of these models from carrying accurate results.

In the proposed model, we adopt a genetic programming approach [3], to derive mathematical expressions for effort estimation applying data mining techniques. A genetic programming system can mine useful relations from past software project data and automatically derive a mathematical model for the estimation of future projects. Therefore, we classify this approach as an analogy method, since it uses analogous data from the past. However, since the system output is in the form of a mathematical expression, it can also get classified as a model-based approach. This duality can be very attractive for the software project effort estimation domain, combining both the data mining's search ability, and the genetic programming symbolic regression's expression ability. The expected outcome of such process is expected to be a strong but also short regression formula, derived by analogy, which will be simple to apply.

The genetic programming paradigm is applied in this research to two software engineering domains, focusing to the estimation of the required effort in two software projects. Several data mining models have been applied recently in software engineering, carrying competitive results that can effectively support project managers [4] [5]. In [6], a case-based reasoning approach is tested and compared with a number of regression models. In [7], a genetic algorithm approach succeeds in producing a quality set of rules for a decision-making framework. In [8], a genetic programming approach has been examined for the assessment of human evaluators in software project management.

The paper is organized as follows. Next section includes the background, presenting the effort estimation concept for software projects and the genetic programming principle. In section 3, the design and the implementation of the genetic programming system is presented. The results and a followed discussion are shown in Section 4. The paper concludes, including with a description of future research, in Section 5.

## 2 Background

### 2.1 Estimating Labor Effort in Software Project Development

Software projects are unique among common engineering projects, as the labor effort affects dramatically the overall project cost. Therefore, an accurate estimation of the effort required for the software development is a decisive factor for the project's success. As a generic principle, in order to estimate the software development effort one has to define the required resources to develop, verify and validate the software product as well as to manage these activities. It is also required to quantify the uncertainty and the risk of this estimation, in the cases that this can be useful [9].

We can classify the related methods into four types: historical analogy, experts' decision, use of models and rules-of-thumb.

- Historical analogy is applied when we have available data from the past that concern similar projects. It mostly involves comparison, using ad-hoc measures and/or data that have been recorded in past projects. These estimations are made for both the high-level overall effort, and for individual tasks while calculating the main cost estimates. The high-level analogy is used in the early phases of the project's life cycle, and it naturally demands further calculation afterwards, since there is rarely a perfect project match.
- Experts' decision uses estimates produced by a human expert based on past project experience. This estimation can be highly subjective, though decently accurate when the expert has mastered both the software domain and the estimation procedure [10].
- The use of models concerns estimates derived using mathematical or parametric cost models. These approaches use empirical equations that have been produced primarily with statistical techniques. Most often they calculate human effort, cost and schedule.
- Rules-of-thumb can also be used for effort estimation. These rules can have various forms and in most cases they involve a very simple mathematical formula, or a percentage allocation of effort over activities or phases, based on historical data.

Usually the task of effort estimation is assigned to a combination of the above techniques, and the contribution of each approach depends on the phase of the software project. During the first stages, estimation is performed mainly by high-level analogies and model-based estimates. In later stages, the required effort becomes more tangible, and the main method used for estimation is the analogy, whereas the model-based estimates are used for sanity-check.

## 2.2 Genetic Programming

Evolutionary computation has already gain respect among scientific community as an effective search and data mining method. Nowadays, a large number of applications are using evolutionary algorithms in the real world domain. These techniques are usually used in domains where a direct search method (e.g. back-propagation in neural networks) cannot be applied or is inefficient due to the nature of the problem. Genetic programming belongs to the evolutionary algorithms and it uses standard genetic operators as crossover, mutation and reproduction [3]. Recent development incorporates to genetic programming special mutation types, such as *shrink mutation* [11], which are shown to provide better search in the solution space. The algorithmic approach used in this work, for training the system using genetic programming, includes six (6) steps:

1. Generate a population of individuals (programs) randomly using the primary expressions provided.
2. Evaluate each individual, assigning a value for its fitness using a problem-specific function, which actually mirrors the suitability of the individual to solve the problem.
3. Use elitism (direct reproduction) to copy existing individuals into a new generation.
4. Recombine genetically the new population, using the crossover operator, from a guided stochastic selection of parents.
5. Mutate genetically the new population using the mutation operators, selecting randomly mutation candidates.
6. Go to step 2, until the termination criterion has been fulfilled.

Each node of an individual can be either a function or the terminal. The function set (FS) contains primitive functions like addition and multiplication, and the terminal set (TS) contains constants and input attributes. As in most GP implementations that address symbolic regression problems, in our work we used  $FS = \{+, -, *, \% \}$  where the symbol “%” means protected division [3]. The real-valued constants used in the system belong to the set  $[-1, 1]$ .

## 3 Design and Implementation

### 3.1 Data Preprocessing

The proposed methodology was applied in two software engineering data sets: the COCOMONASA and the COC81. The COCOMONASA domain has been also examined in past research [12], [13] and [14], while the COC81 data set has been tested

in [15], [12]. These data sets are publicly available in the PROMISE repository<sup>1</sup> that contains software engineering data sets. For the COCOMONASA data, we substituted the original descriptions of the initial data set with numerical values, i.e. the value set *{Very\_Low, Low, Nominal, High, Very\_High}* was substituted by the set *{0,1,2,3,4}*, with the value 0 corresponding to *Very\_Low* and so on. In both data sets, we performed *linearization* to the *Lines-of-Code (KSLLOC)* and *months* features (output feature), by substituting the original values with their natural logarithms. This handling of data was applied based on the conclusions in [12]. After substitution and linearization, we standardized the data inside the range [-1,1], as this treatment has shown in our experiments that it can further improve the search process. In addition, as previously stated, the constants used in a program belong to that range as well. To standardize, for each feature  $\mathbf{y}$ , the equation that follows is applied:

$$_N(y_i) = 2 \cdot \frac{y_i - m_y}{r_y} \quad (1)$$

where:

$_N(y_i)$ : standardized value of  $y_i$

$$m_y = \frac{y_{\max} + y_{\min}}{2} \quad (2)$$

$$r_y = y_{\max} - y_{\min} \quad (3)$$

In Table 2 and Table 3, we summarize the available features and their value ranges for the aforementioned data sets.

### 3.2 Algorithm Setup

The genetic programming system used a *steady-state* genetic process [16]. From the In order, four types available to create the initial population (Variable, Grow, Ramped and Ramped Half and Half) the latter, developed by Koza [3], is used in the majority of the genetic software; therefore we applied it in our research. We used the *tournament selection* [17] as this is the most widely accepted among the evolutionary implementations. The number to randomly select individuals for each tournament is usually 5 to 7. In this paper, we used a group of 7 individuals. Aiming at the improvement of the search process and solution size controlling, we applied a recent adaptive scheme for the operation rates [18], with initial values set to *crossover* 80 % of the time, *mutation* 15% of the time, and *straight copy* 5% of the time. Mutation was further subdivided into 60% *shrink mutation*, 20% *node mutation* and 20% *constant mutation* focusing this way on searching, when possible, small candidate solutions. The crossover applied was a *subtree-crossover*, where two internal nodes are selected in each parent tree and the subtree defined by the first internal node exchanges place with the subtree of the second internal node, if and only if the size for each derived offspring is not larger than the maximum allowed size. This maximum

---

<sup>1</sup> <http://promise.site.uottawa.ca/SERepository/>

*tree size* was set to 650 nodes. These genetic programming parameters are summarized in Table 1.

We performed 10-fold cross validation, since the use of only one sample as test set is susceptible to overfitting [19]. In this methodology, we keep each time separately a 10% of the data to be used as a test set. In general, cross validation can increase the reliability of the results in a regression system, and the suggested number of folds is 5 to 10. During cross validation, each fold is used as the testing data and the rest  $n-1$  folds of data are used as training data to retrain the model and generate evaluation results. The final evaluation outcome is aggregated from the result of each fold. To further improve the search process, we separated this training data into two sets: an actual data set used for the training (called hereinafter as *actual training data set*) and a *validation set*. During the run, the actual training data set is used to evaluate candidate solutions. However, in order to promote a candidate as the solution of the run, in this approach we require that this candidate achieves higher regression score in the validation set as well. This approach can assist in encountering overfitting problems that appear with the use of only a single training set [20].

### 3.3 Fitness Function

The selected fitness measure is *root mean square error (RMSE)*. For the specific problems though, a variance of other measures has also been proposed [6]. Hence, for comparison reasons, we calculate other metrics too, such as the *mean absolute error (MAE)*, the *mean magnitude relative error (MMRE)*, the *PRED(25)* and the *PRED(30)* that have been proposed in [21]. *PRED(r)* calculates the percentage of the estimated values that have relative error less than  $r$ . In past works, *PRED(r)* has been applied with  $r=30$  for the domains of our study, and is also included here. In software engineering, the standard criterion to consider a model acceptable is  $MMRE \leq 0.25$  and  $PRED(25) \geq 75\%$  [22].

**Table 1.** Genetic programming parameter setup

Parameter	Value
Population	9,000 individuals
GP implementation	Steady state GP
Selection	Tournament with elitist strategy
Tournament size	7
Crossover rate	0.8 (adaptive; see [18])
Overall mutation rate	0.15 (adaptive; see [18])
Straight copy rate (Elitism)	0.05 (adaptive; see [18])
Mutation: Shrink mutation rate	0.6
Mutation: Node mutation rate	0.2
Mutation: Constant mutation rate	0.2
Maximum size of individuals (nodes)	650
Maximum number of generations	200

We also included measures which are variance-independent, such as the *root relative square error (RRSE)* and the *root absolute error (RAE)*, in order to enable comparison with future works, since the data in our system has been standardized in the

interval [-1,1] and hence the *RMSE* values cannot be used directly for comparison, unless the same standardization is applied beforehand. The equations that follow summarize the calculation of each aforementioned measure.

$$RMSE = \frac{1}{n} \sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2 \quad (4)$$

$$MAE = \frac{1}{n} \sum_{i=0}^{n-1} |y_i - \hat{y}_i| \quad (5)$$

$$RRSE = \sqrt{\sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2 / \sum_{i=0}^{n-1} (y_i - \bar{y}_i)^2} \quad (6)$$

$$RAE = \frac{\sum_{i=0}^{n-1} |y_i - \hat{y}_i|}{\sum_{i=0}^{n-1} |y_i - \bar{y}_i|} \quad (7)$$

$$MMRE = \frac{100}{n} \sum_{i=0}^{n-1} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (8)$$

$$PRED(r) = \frac{100}{n} \sum_{i=0}^{n-1} \begin{cases} 1 & \text{if } \left| \frac{y_i - \hat{y}_i}{y_i} \right| \leq \frac{r}{100} \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

where:

$y_i$  : actual value of case  $i$

$\hat{y}_i$  : estimated value of case  $i$

$\bar{y}_i$  : mean value of test set cases

$n$  : number of cases in test set

$r$  : value (range) for which the *PRED* function is calculated, usually set to 25 or 30

Having discussed the system design, in the following session we shall apply the methodology in the software engineering domain.

## 4 Results

### 4.1 COCOMONASA Domain

The COCOMONASA domain contains data from 60 NASA projects, from different centers for projects, which were carried during the 1980s and 1990s. This data concerns projects from the aerospace software domain. The available attributes are 17

and they are all numeric: 15 attributes are the effort multipliers, one is the Lines-of-Code (LOC) and one attribute is the actual development effort. The LOC variable has been estimated directly or computed beforehand, using function point analysis [23]. The task is to produce a new cost model, for a given background knowledge. In [12], a very simple calibration method (called COCONUT) achieved PRED(30)=70% and PRED(20)=50%. These results were derived after 30 repeats of an incremental cross-validation process. In the same paper, two cost models are compared; one based in lines of code and one using additionally 14 so-called effort multipliers. The use of only lines of code resulted into the loss 10 to 20 PRED(r) points. In [13], a feature subset selection (FSS) is applied to this software effort data. The paper shows that FSS can dramatically improve cost estimation. Table 2 summarizes the available features and their value ranges. Further details on each feature can be found in [1].

**Table 2.** Data Features and Value Range for COCOMONASA domain

Variable	Description	Max	Min
<i>rely</i>	Required software reliability	4	1
<i>data</i>	Data base size	4	1
<i>cplx</i>	Process complexity	5	1
<i>time</i>	Time constraint for CPU	5	2
<i>stor</i>	Main memory constraint	5	2
<i>virt</i>	Machine volatility	3	1
<i>turn</i>	Turnaround time	3	1
<i>acap</i>	Analysts capability	4	2
<i>aexp</i>	Application experience	4	2
<i>pexp</i>	Programmers capability	4	2
<i>vexp</i>	Virtual machine experience	3	1
<i>lexp</i>	Language experience	3	0
<i>modp</i>	Modern programming practices	4	1
<i>tool</i>	Use of software tools	4	0
<i>sced</i>	Schedule constraint	3	1
<i>ln(KSLOC)</i>	Software size lines-of-code	6.04	0.788
<i>ln(months)</i>	Effort in months	8.08	2.128

Table 3 summarizes our results per fold run, and includes the mean and the standard deviation for each measure and feature of the solution. The column *Generation* is the generation in which the solution was found, and the *Size* column is the number of nodes of the solution tree (e.g. the complexity of the derived mathematical formula). As it can be seen from Table 3, the derived solutions can vary significantly in their size, depending on the fold used. The following solution that was derived in fold #7, is surprisingly small, with only two features used (apart *KSLOC*), and it achieved 100% *PRED(25)*.

$${}_N(\ln(\text{months})) = {}_N(\ln(\text{KSLOC})) - 0.03 \cdot [{}_N(virt) + {}_N(turn)] \quad (10)$$

where  ${}_N(\cdot)$  denotes that the normalized values of the corresponding variables are used.

**Table 3.** GP 10-Fold Cross Validation Results for COCOMONASA domain

Fold #	RMSE	MAE	RRSE	RAE	MMRE	PRED(25)	PRED(30)	Size	Generation
1	0.088	0.071	0.202	0.186	0.158	66.7%	100.0%	511	14
2	0.029	0.026	0.077	0.080	0.084	100.0%	100.0%	313	13
3	0.093	0.082	0.195	0.191	0.189	66.7%	83.3%	467	46
4	0.148	0.105	0.653	0.510	0.302	50.0%	66.7%	73	135
5	0.276	0.181	0.617	0.458	0.328	50.0%	50.0%	301	148
6	0.061	0.040	0.160	0.120	0.112	100.0%	100.0%	509	100
7	0.060	0.048	0.104	0.093	0.098	100.0%	100.0%	7	8
8	0.168	0.142	0.416	0.403	0.223	50.0%	66.7%	5	4
9	0.148	0.103	0.317	0.270	0.677	66.7%	66.7%	195	14
10	0.154	0.118	0.349	0.303	0.424	50.0%	50.0%	3	4
Mean	0.122	0.092	0.309	0.261	0.259	70.0%	78.3%	238	49
StdDev	0.072	0.048	0.202	0.154	0.184	21.9%	20.9%	211	57

It is worth to note that *KSLOC* and *turn* variables are also present in all derived models in the work of [13], while the *virt* variable occurs only in 1 of the 10 models. As stated previously, the *months* and *KSLOC* variables used in our study have been changed to the natural logarithms of the original data set values. If we perform the reverse conversion (e.g. de-normalizing), we result into the following simple relation between the original data set values:

$$months = e^{\frac{0.3803 \ln(KSLOC) - 0.03virt - 0.03turn + 0.2949}{0.3358}} \quad (11)$$

In *Table 4*, the occurrence of each feature in the solutions found for all folds is shown.

**Table 4.** Feature frequency in solutions produced for COCOMONASA domain

Variable	Times	Variable	Times
ln(KSLOC)	10	virt	4
aexp	6	pcap	4
rely	5	vexp	4
data	5	tool	4
time	5	sced	4
stor	5	cplx	3
turn	5	acap	3
lexp	5	modp	2

*Table 5* compares our results for *PRED(30)* to those found in literature with best values in bold. As it can be seen, our system achieved a higher *PRED(30)* rate as compared to past works, in both the average resulted value and the highest one produced. On the other hand, in this table we present a high standard deviation in our system.

**Table 5.** PRED(30) Results Comparison for COCOMONASA domain

Publication	Method	Avg.	Std.Dev	Best
[12]	coconut	n/a	n/a	70.0%
[13]	wrapper FSS	76.7%	7.3%*	81.3%
[14]	lsr_num_ln	69.7%	11.1%	n/a
[14]	lsr_em_ln	68.5%	12.5%	n/a
[14]	m5_num_ln	73.5%	10.7%	n/a
[14]	m5_em_ln	69.7%	10.5%	n/a
[14]	m5_em_loc_ln	60.5%	9.6%	n/a
[14]	lsr_em_loc_ln	60.5%	9.6%	n/a
[14]	m5_num_loc_ln	55.3%	11.7%	n/a
[14]	lsr_num_loc_ln	40.8%	11.7%	n/a
[14]	m5_em	41.0%	14.4%	n/a
[14]	m5_num	41.5%	11.6%	n/a
[14]	m5_num_loc	42.0%	8.9%	n/a
[14]	lsr_num_loc	41.2%	12.7%	n/a
[14]	lsr_em_loc_ln	40.2%	8.4%	n/a
[14]	lsr_num	31.0%	12.7%	n/a
[14]	lsr_em	28.7%	8.4%	n/a
This study	genetic programming	<b>78.3%</b>	20.9%	<b>100%</b>

\* best reported value

The reason for this is that we record here the standard deviation of *PRED(30)* encountered during the 10-fold cross validation, which results from evaluating different test data sets (e.g. for each fold validation). On the other hand, in [13] the test sets are selected randomly for each run, allowing for potential set overlapping; in [12] the standard deviation is reported over 30 runs on the *same* data set. Hence, taking into respect the (completely) different test sets encountered, we consider the high value of our system's standard deviation as being a natural result.

## 4.2 COC81 Domain

The COC81 domain contains data from 63 projects. This data comes from a variety of domains such as financial, engineering and science projects. There are also 17 attributes that are all numeric: 15 attributes are the effort multipliers, one is the Lines-of-Code (LOC) and one attribute is the actual development effort. There are no missing attributes. In [15], a variety of methods are examined into a related data set, including neural networks, regression trees, COCOMO and the SLIM model [24]. The neural networks and function-point based prediction models outperformed regression trees, and the latter outperformed COCOMO and the SLIM model. Table 6 summarizes the available features and their value ranges. A detailed description for these features appears in [1]. As it can be seen from Table 7, the derived solutions can vary significantly in their size, depending on the fold used. The following solution that was produced in fold #1, has only one feature used (apart *KSLC*), and it achieves 57.1% *PRED(25)*.

$${}_N(\ln(months)) = {}_N(\ln(KSLC)) - 0.15 \cdot {}_N(virt) \quad (12)$$

where  ${}_N(\cdot)$  is a symbol for the normalized values of the corresponding variables, as previously.

**Table 6.** Data Features and Value Range for COC81 domain

Variable	Description	Maximum	Minimum
<i>rely</i>	Required software reliability	1.400	0.750
<i>data</i>	Data base size	1.160	0.940
<i>cplx</i>	Process complexity	1.650	0.700
<i>time</i>	Time constraint for CPU	1.660	1.000
<i>stor</i>	Main memory constraint	1.560	1.000
<i>virt</i>	Machine volatility	1.300	0.870
<i>turn</i>	Turnaround time	1.150	0.870
<i>acap</i>	Analysts capability	1.460	0.710
<i>aexp</i>	Application experience	1.290	0.820
<i>pcap</i>	Programmers capability	1.420	0.700
<i>vexp</i>	Virtual machine experience	1.210	0.900
<i>lexp</i>	Language experience	1.140	0.950
<i>modp</i>	Modern programming practices	1.240	0.820
<i>tool</i>	Use of software tools	1.240	0.830
<i>sced</i>	Schedule constraint	1.230	1.000
<i>ln(KSLOC)</i>	Software size lines-of-code	7.048	0.683
<i>ln(months)</i>	Effort in months	9.341	1.775

The variable *KSLOC* is also present in all derived models in the work of [12], and the *virt* variable occurs in 9 of the 10 models. As stated previously, the *months* and *KSLOC* variables used in our study have been changed to the natural logarithms of the original data set values. By performing the reverse necessary conversions (e.g. de-standardizing), we conclude to the following simple equation between the original data set values:

$$months = e^{1.188\ln(KSLOC) - 2.637virt + 20.05} \quad (13)$$

Table 8 summarizes the occurrence of each feature to the solutions found in all folds.

**Table 7.** GP 10-Fold Cross Validation Results for COC81 domain

Fold #	RMSE	MAE	RRSE	RAE	MMRE	PRED(25)	PRED(30)	Size	Generation
1	0.174	0.119	0.449	0.337	0.465	57.1%	57.1%	5	2
2	0.147	0.133	0.389	0.408	0.836	42.9%	42.9%	107	8
3	0.214	0.156	0.782	0.647	0.465	50.0%	50.0%	7	3
4	0.206	0.162	0.741	0.761	0.949	50.0%	50.0%	11	8
5	0.294	0.231	0.933	0.841	1.895	33.3%	33.3%	5	6
6	0.221	0.192	0.700	0.734	1.027	50.0%	50.0%	23	8
7	0.305	0.224	0.525	0.441	1.411	50.0%	50.0%	15	10
8	0.219	0.169	0.352	0.332	1.209	50.0%	50.0%	13	7
9	0.208	0.183	0.332	0.351	0.458	50.0%	50.0%	5	13
10	0.201	0.182	0.376	0.403	0.696	42.9%	42.9%	7	5
Mean	0.219	0.175	0.558	0.526	0.941	47.6%	47.6%	20	7
StdDev	0.048	0.035	0.214	0.197	0.467	6.4%	6.4%	31	3

Table 9 compares our results for *PRED(30)* to those found in literature. The best values are shown in bold. The low success rates for all models reflect the fact these *COC81* data concern projects from different domains (e.g. financial, engineering, etc.) while the *COCOMONASA* data addressed the aerospace project domain only, which

follows the *stratification hypothesis* [25]. A comparison to the results of [15] is not included, since in that publication, an extended feature set was used (e.g. 39 attributes were used, instead of the 17 ones that have become publicly available in the PROMISE repository). As it can be observed in the results presented in Table 9, our system outperformed those found in literature, in both the average and best  $PRED(30)$  values, as well as to its standard deviation.

**Table 8.** Feature frequency in solutions produced for COC81 domain

Variable	Times	Variable	Times
ln(KSLOC)	10	cplx	2
virt	6	acap	2
stor	4	turn	1
rely	3	time	1
vexp	2	pcap	1
tool	2	aexp	1
sced	2	modp	0
lexp	2	data	0

**Table 9.**  $PRED(30)$  Results Comparison for COC81 domain

Publication	Method	Average	Std. dev	Best
[13] *	wrapper FSS	45.8%	9.3%	51.3%
[14]	lsr_num_ln	44.3%	10.8%	n/a
[14]	lsr_em_ln	40.0%	9.7%	n/a
[14]	m5_num_ln	39.7%	13.7%	n/a
[14]	m5_em_ln	38.4%	9.2%	n/a
[14]	m5_em_loc_ln	21.7%	8.5%	n/a
[14]	lsr_em_loc_ln	21.7%	8.5%	n/a
[14]	m5_num_loc_ln	20.6%	6.9%	n/a
[14]	lsr_num_loc_ln	20.6%	6.9%	n/a
[14]	m5_em	15.4%	8.4%	n/a
[14]	m5_num	13.7%	8.7%	n/a
[14]	m5_num_loc	11.7%	6.9%	n/a
[14]	lsr_num_loc	11.3%	6.7%	n/a
[14]	lsr_em_loc_ln	11.3%	6.7%	n/a
[14]	lsr_num	9.4%	6.7%	n/a
[14]	lsr_em	7.9%	6.8%	n/a
[26]*	coc81:kind.max	47%	51.0%	n/a
This study	Genetic programming	<b>47.6%</b>	<b>6.4%</b>	<b>57.1%</b>

\* best reported value

## 5 Conclusions and Further Research

Nowadays, it is acknowledged that there is a need for accurate and easily applicable models for effort estimation in software projects. In this paper, we presented a genetic programming system for symbolic regression that can be applied in effort estimation tasks. The system involved a data preprocessing phase aiming to enhance the search process. This model was further refined by the incorporation of a cross-validation technique and the use of a validation set. We paid attention to the selected genetic operators and their corresponding rates in order to make search more effective and

efficient and at the same moment to maintain solution comprehensibility. The proposed model was then effectively tested in two software engineering domains that have recently become publicly available from the PROMISE data repository. In both cases, the system was shown able to derive results that not only achieve higher regression accuracy as compared to those found in literature, but also are in the form of easily interpretable mathematical formulas, ready to be used by project managers. The overall methodology was shown fairly robust to be tested into more software engineering estimation domains.

Further work will involve the implementation of this genetic programming model into more software engineering problems, such as defect prediction and text mining tasks, aiming to analyze code comprehensibility. Moreover, we plan to use this system using incrementally smaller test sets, to enable conclusions on incremental hold-out results [12]. Finally, we aim to incorporate strongly typed genetic programming [27] to evolve complex intelligent structures, such as *Takagi-Sugeno* fuzzy rule-based systems [28]. We expect that a strongly typed approach will further improve the resulted precision, since it will restrict the subspace of rational solutions.

## References

1. Boehm, B.: Software Engineering Economics. Prentice-Hall, Englewood Cliffs (1981)
2. Price, S. (2007), <http://www.pricesystems.com>
3. Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge (1992)
4. Rodríguez, D., Cuadrado, J.J., Sicilia, M.A., Ruiz, R.: Segmentation of Software Engineering Datasets Using the M5 Algorithm. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3994, pp. 789–796. Springer, Heidelberg (2006)
5. Menzies, T., Di Stefano, J.S.: How Good is your Blind Spot Sampling Policy? In: Proc. of 8th IEEE Int'l Symp. on High Assurance Systems Eng., Tampa, FL, USA (2004)
6. Shepperd, M., Schofield, C.: Estimating software project effort using analogies. IEEE Trans. Soft. Eng. 23(12) (1997)
7. Aguilar-Ruiz, J.S., Ramos, I., Riquelme, J.C., Toro, M.: An evolutionary approach to estimating software development projects. Information and Software Technology 43, 875–882 (2001)
8. Boetticher, G., Lokhandwala, N., Helm, J.C.: Understanding the Human Estimator. In: 2nd Int'l. Predictive Models in Soft. Eng (PROMISE) Workshop, 22nd IEEE Int'l. Conf. on Soft. Maintenance, PA, USA (2006)
9. Lum, K., Bramble, M., Hihn, J., Hackney, J., Khorrami, M., Monson, E.: Handbook of Software Cost Estimation, Jet Propulsion Laboratory, Pasadena, CA, USA (2003)
10. Hihn, J., Habib-agathi, H.: Cost estimation of Software Intensive Projects: A survey of Current Practices. In: Proc. of the 13<sup>th</sup> IEEE Int'l. Conf. Soft. Eng. (1991)
11. Singleton, A.: Genetic Programming with C++. BYTE Magazine, February Issue (1991)
12. Menzies, T., Port, D., Chen, Z., Hihn, J., Stukes, S.: Validation Methods for Calibrating Software Effort Models. In: Proc. ICSE 2005, St. Louis, MI, USA (2005)
13. Chen, Z., Menzies, T., Port, D., Boehm, B.: Feature Subset Selection Can Improve Software Cost Estimation Accuracy. In: Proc. 1st Int'l. Predictive Models in Soft. Eng. (PROMISE) Workshop St. Louis, MI, USA (2005)

14. Menzies, T., Chen, D.P.Z., Hihn, H.: Simple Software Cost Analysis: Safe or Unsafe? In: Proc. 1<sup>st</sup> Int'l. Predictive Models in Soft. Eng. (PROMISE) Workshop, St. Louis, MI, USA (2005)
15. Srinivasan, K., Fisher, D.: Machine Learning Approaches to Estimating Software Development Effort. *IEEE Trans. Soft. Eng.* 21(2), 126–137 (1995)
16. Rogers, A., Prügel-Bennett, A.: Modeling the dynamics of steady-state genetic algorithms. In: Banzhaf, W., Reeves, C. (eds.) *Foundations of Genetic Algorithms*, pp. 57–68. Morgan Kaufmann, San Francisco (1999)
17. Blickle, T., Theile, L.: A mathematical analysis of tournament selection. In: Eshelman, L.J. (ed.) *Proc. of the 6<sup>th</sup> International Conference on Genetic Algorithms*, pp. 9–16. Lawrence Erlbaum Associates, Hillsdale (1995)
18. Tsakonas, A., Dounias, G.: Evolving Neural-Symbolic Systems Guided by Adaptive Training Schemes: Applications in Finance. *Applied Artificial Intelligence* 21(7), 681–706 (2007)
19. Eads, D., Hill, D., Davis, S., Perkins, S., Ma, J., Porter, R., Theiler, J.: Genetic Algorithms and Support Vector Machines for Time Series Classification. In: *Proc. SPIE*, vol. 4787, pp. 74–85 (2002)
20. Quinlan, J.R.: Bagging, boosting, and C4.5. In: *Proc. 13<sup>th</sup> Nat. Conf. Art. Intell.*, pp. 725–730 (1996)
21. Conte, S.D., Dunsmore, H.E., Shen, V.: *Software Engineering Metrics and Models*. Benjamin-Cummings (1986)
22. Dolado, J.J.: On the problem of the software cost function. *Information and Software Technology* 43, 61–72 (2001)
23. Dreger, J.: *Function Point Analysis*. Prentice Hall, Englewood Cliffs (1989)
24. Putnam, L.H.: A general empirical solution to the macro software sizing and estimating problem. *IEEE Trans. Soft. Eng.* 4(4), 345–361 (1978)
25. Boehm, B., Horowitz, E., Madachy, R., Reifer, D., Clark, B.K., Steece, B., Brown, A.W., Chulani, S., Abts, C.: *Software Cost Estimation* (2000)
26. Menzies, T., Chen, Z., Hihn, J., Lum, K.: Selecting Best Practices for Effort Estimation. *IEEE Transactions Software Engineering* 32(11) (November 2006)
27. Montana, D.J.: Strongly Typed Genetic Programming. *Evolutionary Computation* 3(2) (1995)
28. Takagi, T., Sugeno, M.: Fuzzy Identification of Systems and its Application to Modeling and Control. *IEEE Trans. On Systems, Man and Cybernetics* 17, 295–301 (1985)

# Extracting and Rendering Representative Sequences\*

Alexis Gabadinho, Gilbert Ritschard, Matthias Studer, and Nicolas S. Müller

Department of Econometrics and Laboratory of Demography, University of Geneva  
40, bd du Pont-d'Arve, CH-1211 Geneva, Switzerland  
[alexis.gabadinho@unige.ch](mailto:alexis.gabadinho@unige.ch)  
<http://mephisto.unige.ch/TraMineR>

**Abstract.** This paper is concerned with the summarization of a set of categorical sequences. More specifically, the problem studied is the determination of the smallest possible number of representative sequences that ensure a given coverage of the whole set, i.e. that have together a given percentage of sequences in their neighbourhood. The proposed heuristic for extracting the representative subset requires as main arguments a pairwise distance matrix, a representativeness criterion and a distance threshold under which two sequences are considered as redundant or, identically, in the neighborhood of each other. It first builds a list of candidates using a representativeness score and then eliminates redundancy. We propose also a visualization tool for rendering the results and quality measures for evaluating them. The proposed tools have been implemented in our TraMineR R package for mining and visualizing sequence data and we demonstrate their efficiency on a real world example from social sciences. The methods are nonetheless by no way limited to social science data and should prove useful in many other domains.

**Keywords:** Categorical sequences, Representatives, Pairwise dissimilarities, Discrepancy of sequences, Summarizing sets of sequences, Visualization.

## 1 Introduction

In the social sciences, categorical sequences appear mainly as ordered list of states (employed/unemployed) or events (leaving parental home, marriage, having a child) describing individual life trajectories, typically longitudinal biographical data such as employment histories or family life courses. One widely used approach for extracting knowledge from such sets consists in computing pairwise distances by means of sequence alignment algorithms, and next clustering the sequences by using these distances [1]. The expected outcome of such a strategy is a typology, with each cluster grouping cases with similar patterns (trajectories). An important aspect of sequence analysis is also to compare the patterns of cases grouped according to the values of covariates (for instance sex or socioeconomic position in the social sciences).

A crucial task is then to summarize groups of sequences by describing the patterns that characterize them. This could be done by resorting to graphical representations

---

\* This work is part of the Swiss National Science Foundation research project FN-122230 “Mining event histories: Towards new insights on personal Swiss life courses”.

such as sequence index plots, state distribution plots or sequence frequency plots [2]. However, relying on these graphical tools suffers from some drawbacks. The summarizing task is mainly subjective and is rapidly complicated when there is a great number of distinct patterns, as is often the case.

Hence, we need more appropriate tools for extracting the key features of a given subset or data partition. We propose an approach derived from the concept of representative set used in the biological sciences [3,4]. The aim in this field is mainly to get a reduced reference base of protein or DNA sequences for optimizing the retrieval of a recorded sequence that resembles to a provided one. In this setting, the representative set must have “maximum coverage with minimum redundancy” i.e. it must cover all the spectrum of distinct sequences present in the data, including “outliers”.

Our goal is similar regarding the elimination of redundancy. It differs, however, in that we consider in this paper representative sets with a user controlled coverage level, i.e. we do not require maximal coverage. We thus define a representative set as a set of non redundant “typical” sequences that largely, though not necessarily exhaustively covers the spectrum of observed sequences. In other words, we are interested in finding a few sequences that together summarize the main traits of a whole set.

We could imagine synthetic — not observed — typical sequences, in the same way as the mean of a series of numbers that is generally not an observable individual value. However, the sequences we deal with in the social sciences (as well as in other fields) are complex patterns and modeling them is difficult since the successive states in a sequence are most often not independent of each other. Defining some virtual non observable sequence is therefore hardly workable, and we shall here consider only representative sets constituted of existing sequences taken from the data set itself.

Since this summarizing step represents an important data reduction, we also need indicators for assessing the quality of the selected representative sequences. An important aspect is also to visualize these in an efficient way. Such tools and their application to social science data are presented in this paper. These tools are included in our TraMineR library for mining and visualizing sequences in R [5].

## 2 Data

To illustrate our purpose we consider a data set from [6] stemming from a survey on transition from school to work in Northern Ireland. The data contains 70 monthly activity state variables from July 1993 to June 1999 for 712 individuals. The alphabet is made of 6 states: EM (Employment), FE (Further education), HE (Higher education), JL (Joblessness), SC (School) and TR (Training).

The three first sequences of this data set represented as distinct states and their associated durations (the so called State Permanence Format) look as follows

Sequence

- [1] EM/4-TR/2-EM/64
- [2] FE/36-HE/34
- [3] TR/24-FE/34-EM/10-JL/2

We consider in this paper the outcome of a cluster analysis of the sequences based on Optimal Matching (OM). The OM distance between two sequences  $x$  and  $y$ , also known

as edit or Levenshtein distance, is the minimal cost in terms of indels — insertions and deletions — and substitutions necessary to transform  $x$  into  $y$ . We computed the distances using a substitution cost matrix based on transition rates observed in the data set and an indel cost of 1. The clustering is done with an agglomerative hierarchical method using the Ward criterion. A four cluster solution is chosen. Table 1 indicates some descriptive statistics for each of them. The clusters define four subsets grouping sequences with "similar" patterns, but to interpret the results we need to summarize their content, that is to do cluster labelling.

The sequence frequency plots in Fig. 1 display the 10 most frequent sequences in each cluster and give a first idea of their content. The bar widths are proportional to the sequence frequencies. The 10 most frequent sequences represent about 40% of all the sequences in cluster 1 and 2, while this proportion is 27% and 21% for clusters 3 and 4 due to a higher diversity of the patterns.

### 3 Extracting Representative Subsets

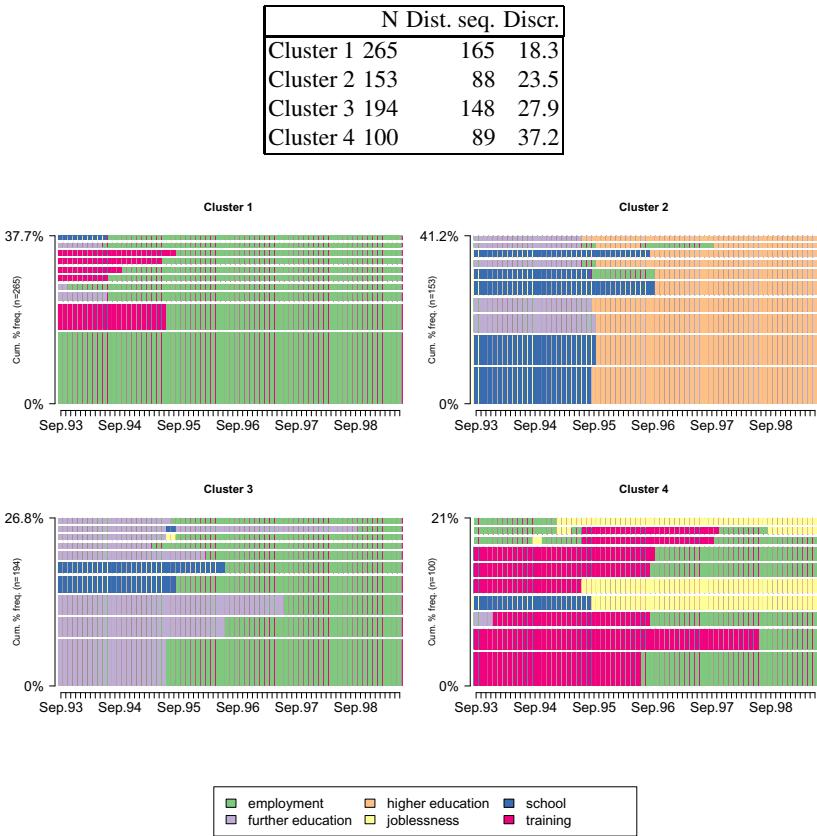
Our main aim is to find a small subset of non redundant sequences that ensures a given coverage level, this level being defined as the percentage of cases that are within a given neighbourhood of at least one of the representative sequences. We propose an heuristic for determining such a representative subset.

It works in two main steps. In the first stage it prepares a sorted list of candidate representative sequences without caring for redundancy and eliminates redundancy within this list in a second stage. It basically requires the user to specify a representativeness criterion for the first stage and a similarity threshold for evaluating redundancy in the second one. The strategy for selecting the sequences that will form the representative set can be summarized as follow:

1. Compute a representativeness score for each distinct sequence in the data set according to the selected criterion;
2. Sort all distinct sequences according to their score, in ascending order if the criterion increases with representativeness or in descending order otherwise;
3. Select a rule for possibly limiting the size of the candidate list;
4. Remove iteratively from the candidate list each sequence that has dissimilarity below a given threshold with any already retained sequence. This ensures that the final set of representative sequences does not contain any pair of sequences that are closer from each other than the predefined threshold.

#### 3.1 Sorting Candidates

The initial candidate list is made of all distinct sequences appearing in the data. Since the second stage will extract non redundant representative sequences sequentially starting with the first element in the list, sorting the candidates according to a chosen representativeness criterion ensures that the "best" sequences given the criterion will be included. We present here four alternatives for measuring the sequence representativeness. The first three measures, *neighbourhood density*, *centrality* and *frequency* are directly obtained from the distance matrix itself, while the fourth is obtained by statistical

**Table 1.** Number of cases, distinct sequences and discrepancy within each cluster**Fig. 1.** 10 most frequent sequences within each cluster

modeling. Alternative vector of representativeness scores can also be provided by the user.

**Neighbourhood Density.** This criterion is the number — the density — of sequences in the neighbourhood of each candidate sequence. This requires to set the neighbourhood radius. We suggest to set it as a given proportion of the maximal theoretical distance between two sequences. Sequences are sorted in decreasing density order. This criterion leads indeed to sort the candidates according to their potential coverage. The neighbourhood density for each sequence in the set is obtained from the distance matrix by counting by row or column the number of distances that are less than a defined threshold (the neighbourhood radius).

**Centrality.** A classical representative of a data set used in cluster analysis is the *medoid*. It is defined as the most central object, i.e. the one with minimal sum of distances to all other objects in the set [7]. This leads to use the sum of distances to all

other sequences, i.e. the centrality as a representativeness criterion. The smallest the sum, the most representative the sequence. It may be mentioned that the most central observed sequence is also the nearest from the ‘virtual’ true center of the set [8]. The centrality of each sequence in the set is obtained from the distance matrix by summing the distances by row or column.

**Frequency.** Sequences may also be sorted according to their frequency. The more frequent a sequence the more representative it is supposed to be. Hence, sequences are sorted in decreasing frequency order. This criterion makes sense in sets where some or all sequences appear more than once. This is indeed the density criterion with the neighbourhood radius set to 0. The frequency of each sequence in the set is obtained from the distance matrix by counting by row or column the distances that are equal to 0 (a distance of 0 between two sequences meaning that they are identical).

**Likelihood.** The sequence likelihood  $P(s)$  is defined as the product of the probability with which each of its observed successive state is supposed to occur at its position. Let  $s = s_1 s_2 \cdots s_\ell$  be a sequence of length  $\ell$ . Then

$$P(s) = P(s_1, 1) \cdot P(s_2, 2) \cdots P(s_\ell, \ell)$$

with  $P(s_t, t)$  the probability to observe state  $s_t$  at position  $t$ . The question is how to determinate the state probabilities  $P(s_t, t)$ . One commonly used method for computing them is to postulate a Markov model, which can be of various order. Below, we just consider probabilities derived from the first order Markov model, that is each  $P(s_t, t)$ ,  $t > 1$  is set to the transition rate  $p(s_t | s_{t-1})$  estimated across sequences from the observations at positions  $t$  and  $t - 1$ . For  $t = 1$ , we set  $P(s_1, 1)$  to the observed frequency of the state  $s_1$  at position 1. The likelihood  $P(s)$  being generally very small, we use  $-\log P(s)$  as sorting criterion. The latter quantity is minimal when  $P(s)$  is equal to 1, which leads to sort the sequences in ascending order of their score.

### 3.2 Eliminating Redundancy

Once a sorted list of candidates has been defined, the second stage consists in extracting a set of non-redundant representatives from the list. The procedure is as follows:

1. Select the first sequence in the candidate list (the best one given the chosen criterion);
2. Process each next sequence in the sorted list of candidates. If this sequence is similar to none of those already in the representative set, that is distant from more than a predefined threshold from all of them, add it to the representative set.

The threshold for redundancy (similarity) is defined as a proportion of the maximal theoretical distance between two sequences and is the same as the neighbourhood radius that is used for computing the coverage (see below) or the neighbourhood density. For the OM distance between two sequences  $(s_1, s_2)$  of length  $(\ell_1, \ell_2)$  this theoretical maximum is

$$D_{max} = \min(\ell_1, \ell_2) \cdot \min(2C_I, \max(S)) + |\ell_1 - \ell_2| \cdot C_I$$

where  $C_I$  is the indel cost and  $\max S$  the maximal substitution cost.

### 3.3 Controlling Size/Coverage Trade-off

Limiting our representative set to the mere sequence(s) with the best representative score may lead to leave a great number of sequences badly represented. Alternatively, proceeding the complete list of candidates to ensure that each sequence in the data set is well represented may not be a suitable solution if we look for a small set of representative sequences.

To control the trade-off between size and representativeness, we use a threshold  $trep$  for the *coverage* level, that is the percentage of sequences having a representative in their neighbourhood. The coverage is recomputed each time that a sequence is added to the representative set and the selection process stops when the coverage threshold is reached.

Alternatively we can set the desired number of representatives and let the coverage unspecified. For example selecting the medoid as representative is done by choosing the *centrality* criterion and setting the number of representatives to 1.

## 4 Measuring Quality

A first step to define quality measures for the representative set is to assign each sequence to its nearest representative according to the considered pairwise distances. Let  $r_1 \dots r_{nr}$  be the  $nr$  sequences in the representative set and  $d(s, r_i)$  the distance between the sequence  $s$  and the  $i$ th representative. Each sequence  $s$  is assigned to its closer representative. When a sequence is equally distant from two or more representatives, the one with the highest representativeness score is selected. Hence, letting  $n$  be the total number of sequences and  $na_i$  the number of sequences assigned to the  $i$ th representative, we have  $n = \sum_{i=1}^{nr} na_i$ . Once each sequence in the set is assigned to a representative, we can derive the following quantities from the pairwise distance matrix.

**Mean Distance.** Let  $SD_i = \sum_{j=1}^{na_i} d(s_j, r_i)$  be the sum of distances between the  $i$ th representative and its  $na_i$  assigned sequences. A quality measure is then

$$MD_i = \frac{SD_i}{na_i}$$

the mean distance to the  $i$ th representative.

**Coverage.** Another quality indicator is the number of sequences assigned to the  $i$ th representative that are in its neighbourhood, that is within a distance  $dn_{max}$

$$nb_i = \sum_{j=1}^{na_i} \left( d(s_j, r_i) < dn_{max} \right).$$

The threshold  $dn_{max}$  is defined as a proportion of  $D_{max}$ . The total coverage of the representative set is the sum  $nb = \sum_i^{nr} nb_i$  expressed as a proportion of the number  $n$  of sequences, that is  $nb/n$ .

**Distance Gain.** A third quality measure is obtained by comparing the sum  $SD_i$  of distances to the  $i$ th representative to the sum  $DC_i = \sum_{j=1}^{na_i} d(s_j, c)$  of the distances of

each of the  $na_i$  sequences to the center of the complete set. The idea is to measure the gain of representing those sequences by their representative rather than by the center of the set. We define thus the quality measure  $Q_i$  of the representative sequence  $r_i$  as

$$Q_i = \frac{DC_i - SD_i}{DC_i}$$

which gives the relative gain in the sum of distances. Note that  $Q_i$  may be negative in some circumstances, meaning that the sum of the  $na_i$  distances to the representative  $r_i$  is higher than the sum of distances to the true center of the set. A similar measure can be used to assess the overall quality of the representative set, namely

$$Q = \frac{\sum_i^{nr} DC_i - \sum_i^{nr} SD_i}{\sum_i^{nr} DC_i} = \sum_{i=1}^{nr} \frac{DC_i}{\sum_j DC_j} Q_i .$$

Representing all sequences by a sequence located exactly at the center of the set yields  $Q = 0$ .

**Discrepancy.** A last quality measure is the sum  $SC_i = \sum_{j=1}^{na_i} d(s_j, c_i)$  of distances to the true center  $c_i$  of the  $na_i$  sequences assigned to  $r_i$ , or the mean of those distances  $V_i = SC_i/na_i$ , which can be interpreted as the discrepancy of the set [8].

## 5 Visualizing Representative Sequences

A graphical tool for visualizing the selected representative sequences together with information measures is included in the TraMineR package. A single function produces a “representative sequence plot” (Figure 2) where the representative sequences are plotted as horizontal bars with width proportional to the number of sequences assigned to them. Sequences are plotted bottom-up according to their representativeness score. Above the plot, two parallel series of symbols associated to each representative are displayed horizontally on a scale ranging from 0 to the maximal theoretical distance  $D_{max}$ . The location of the symbol associated to the representative  $r_i$  indicates on axis  $A$  the (pseudo) variance ( $V_i$ ) within the subset of sequences assigned to  $r_i$  and on the axis  $B$  the mean distance  $MD_i$  to the representative.

**Key Patterns.** The set of representative sequences extracted using the neighbourhood density criterion and a coverage threshold of 25% is displayed in Figure 2 for each of the four clusters of our example. The plots give clearly a more readily interpretable view of the content of the clusters than the frequency plots displayed in Figure 1. Detailed statistics about these sets are presented in Table 2 and overall statistics in Table 3.

The pairwise distances used are the optimal matching distances that we used for the clustering. The threshold  $dn_{max}$  for similarity (redundancy) between sequences was set as 10% of the maximal theoretical distance  $D_{max}$ . The sequence length being

**Table 2.** Representative sequences by cluster, density criterion, coverage=25%

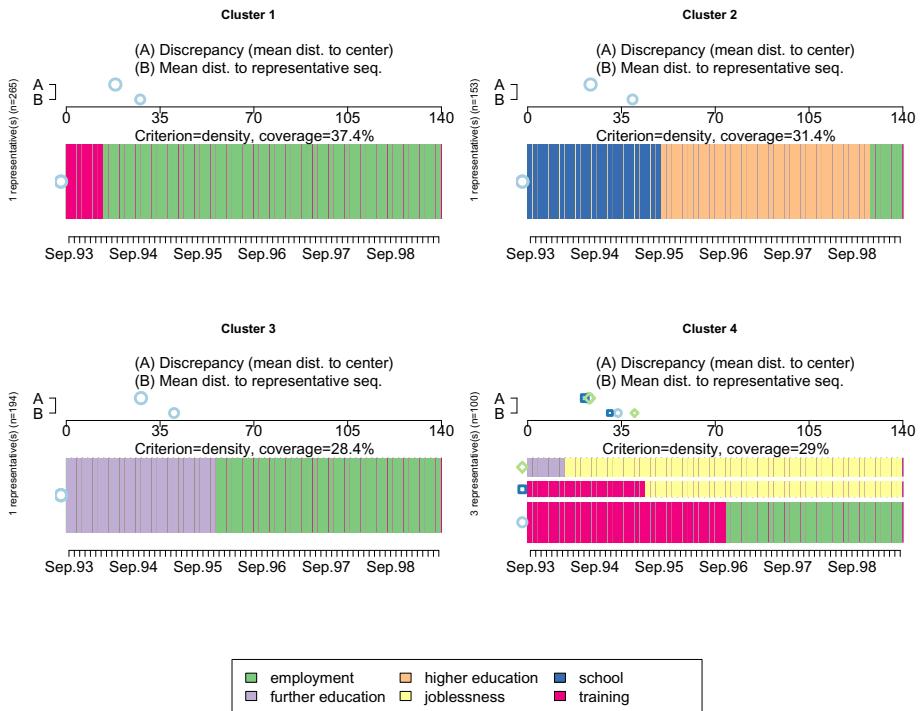
	<i>na</i>	(%)	<i>nb</i>	(%)	<i>MD</i>	<i>V</i>	<i>Q</i>
Cluster 1							
<i>r</i> <sub>1</sub>	265	100.0	99	37.4	27.6	18.3	-50.7
Cluster 2							
<i>r</i> <sub>1</sub>	153	100.0	48	31.4	39.1	23.5	-66.7
Cluster 3							
<i>r</i> <sub>1</sub>	194	100.0	55	28.4	40.2	27.9	-44.3
Cluster 4							
<i>r</i> <sub>1</sub>	54	54.0	16	16.0	33.7	22.7	-0.0
<i>r</i> <sub>2</sub>	21	21.0	7	7.0	30.7	21.3	4.8
<i>r</i> <sub>3</sub>	25	25.0	6	6.0	39.9	23.2	18.8

**Table 3.** Comparing criterions with coverage of 25%, 50% and 75%

	<i>nr</i>	<i>COV</i>	<i>MD</i>	<i>Q</i>	<i>nr</i>	<i>COV</i>	<i>MD</i>	<i>Q</i>	<i>nr</i>	<i>COV</i>	<i>MD</i>	<i>Q</i>
Cluster 1												
Density	1	37.4	27.6	-50.7	2	51.7	21.9	-19.8	12	75.1	14.3	22.1
Frequency	1	28.3	30.5	-66.6	3	53.2	15.6	15.0	20	75.1	8.2	55.3
Likelihood	1	28.3	30.5	-66.6	3	53.2	15.6	15.0	14	75.1	9.9	45.9
Centrality	2	38.9	23.0	-25.5	4	61.1	16.1	11.8	17	75.1	11.9	34.8
Cluster 2												
Density	1	31.4	39.1	-66.7	3	55.6	17.8	24.0	8	75.2	12.0	48.8
Frequency	2	40.5	18.7	20.5	4	52.3	15.0	35.9	9	75.8	8.5	63.6
Likelihood	2	40.5	18.7	20.5	4	51.0	14.7	37.3	9	75.2	8.4	64.1
Centrality	2	26.1	31.1	-32.6	10	64.7	13.4	43.1	15	78.4	9.5	59.6
Cluster 3												
Density	1	28.4	40.2	-44.3	5	51.0	24.0	13.9	28	75.3	11.5	58.9
Frequency	2	32.0	34.4	-23.5	6	51.5	19.3	30.6	34	75.3	9.3	66.6
Likelihood	2	30.4	32.1	-15.2	6	51.5	21.9	21.3	31	75.3	9.9	64.5
Centrality	2	33.0	35.4	-26.8	13	51.5	24.7	11.5	48	75.3	11.4	59.2
Cluster 4												
Density	3	29.0	34.6	7.0	10	51.0	22.7	38.9	33	75.0	10.5	71.7
Frequency	3	26.0	34.0	8.6	18	50.0	19.2	48.4	37	75.0	9.2	75.4
Likelihood	3	27.0	34.7	6.9	11	50.0	22.1	40.5	34	75.0	10.3	72.3
Centrality	14	35.0	30.9	17.0	26	51.0	21.2	42.9	45	75.0	10.6	71.6

$\ell = 70$ , the indel cost 1 and the maximal substitution cost 1.9995, we get  $D_{max} = 70 \cdot \min(2, 1.9995) = 139.96$ .

The first cluster is represented by a sequence begining with a short spell of training followed by employment during the rest of the period. This single representative covers (within 10% of  $D_{max}$ ) 99 sequences (37%) of the cluster (Table 2). Hence, this cluster is characterized by patterns of rapid entry into employment. The overall quality measures

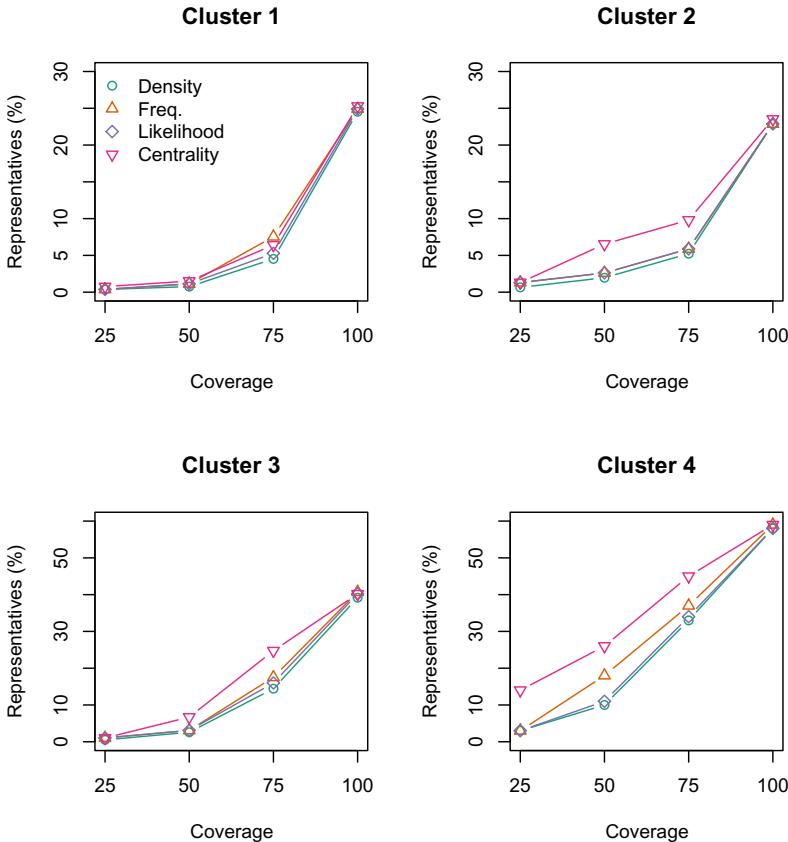


**Fig. 2.** Representative sequences selected with the density criterion, within each cluster - 25% coverage threshold

for the representative set are indeed the same as those for the single representative it contains.

The second cluster is described by a pattern leading to higher education and then to employment, starting with a spell of school. This pattern covers (have in its neighbourhood) 31% of the sequences. In cluster 3, the pattern is a transition to employment preceded by long (compared to Cluster 1) spells of further education.

The key patterns in cluster 4 were less clear when looking at the sequence frequency plot (Figure 1). The diversity of the patterns is high in this cluster which leads to the extraction of three non redundant sequences from the candidate list to achieve the 25% coverage: one is a long spell of training leading to employment and the two others are long spells of joblessness preceded by either a short spell of further education or a long spell of training. Hence these trajectories can be characterized as less successful transitions from school to work. The overall quality measure reaches its highest level ( $Q = 7\%$ ). The discrepancy is high in this group ( $V = 37.2$ ) and the three selected representatives cover the sequence space so that representing the sequences with their assigned representative rather than by the center of the set significantly decreases the sum of distances.

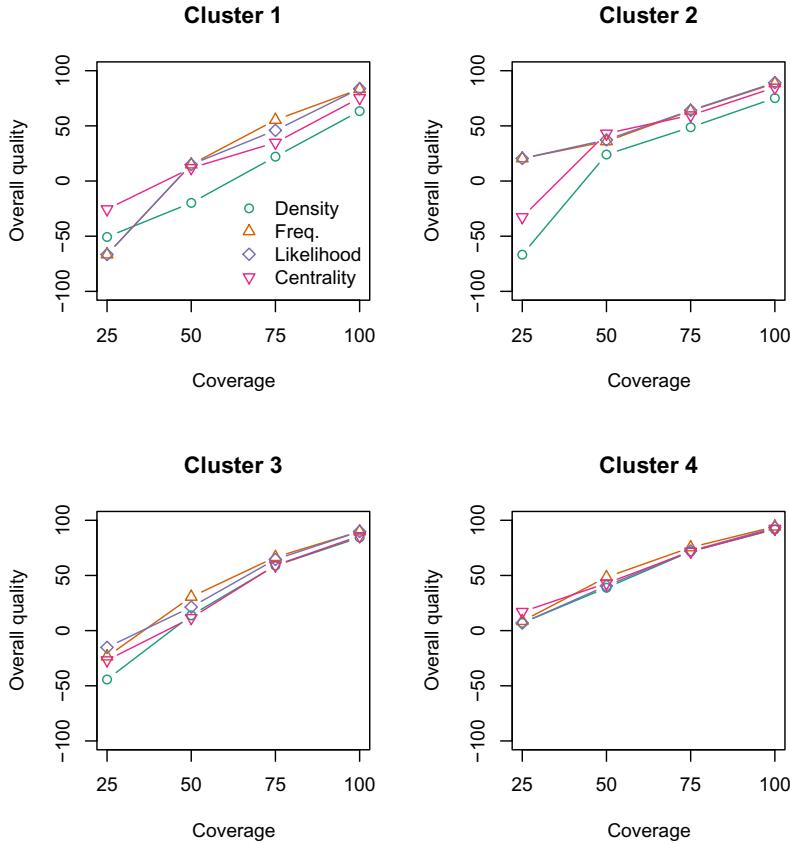


**Fig. 3.** Number of representative sequences (as the percentage of all sequences in the cluster) selected with several criterions, with *trep* of 0.25, 0.50, 0.75 and 1.0

## 6 Comparing Sorting Criterions

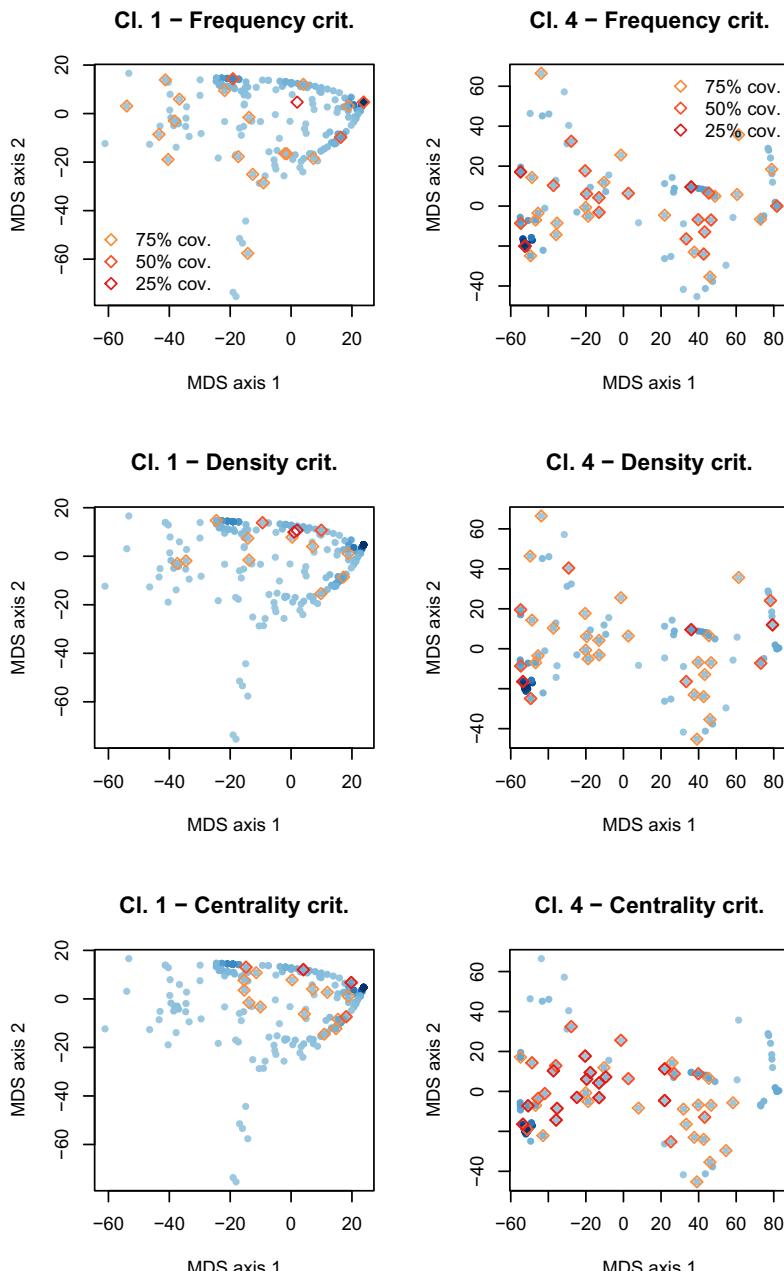
Sorting the candidate list according to the distance to the center yields poor results in many cases, as measured by the number of representatives needed for achieving a given coverage level. Indeed selecting the objects closest from the center of the group leads to poor representation of objects that are far from it, as shown in Figure 5. However, the centrality criterion may yield better overall quality measures with reduced coverage (50% and below). Depending on the spatial distribution of the sequences as defined by the distance matrix, uncovered sequences may indeed be much more distant from their attributed representative than from the center of the set.

The third part of Table 3 presents the results obtained after increasing the *trep* coverage threshold to 75%. As a consequence the proportion of well represented sequences



**Fig. 4.** Overall quality obtained with several criterions, with coverage of 25%, 50%, 75% and 100%

is now at least 75%. This gain comes however at the cost of a considerable increase in the number  $nr$  of selected representative sequences. Full coverage is achieved with about one quarter of the sequences in clusters 1 and 2, while 60% of the sequences are needed in cluster 4 (Figure 3). Table 3 and Figure 4 show how increasing coverage leads to a decrease in mean distance to representative and an increase in overall quality. The mean distance to representative approaches 0 and is below the neighbourhood radius when full coverage is reached, while overall quality approaches 100%. Table 3 and Figure 3 confirm that the neighbourhood density criterion yields systematically the smallest number of representatives for each cluster and coverage level. The best results for the overall quality measure is obtained with the frequency criterion for three of the four clusters. Indeed with the frequency criterion the representatives that have the most sequences having a null distance to them (the highest frequency) are selected first, impacting favourably the overall quality measure.



**Fig. 5.** Multidimensional scaling (MDS) representation of the pairwise distance matrix and selected representatives, with coverage of 25%, 50% and 75% - Cluster 1 and 4

## 7 Conclusions

We have presented a flexible method for selecting and visualizing representatives of a set of sequences. The method attempts to find the smallest number of representatives that achieve a given coverage. Different indicators have been considered to measure representativeness and the coverage can be evaluated by means of different sequence dissimilarity measures. The heuristic can be fine tuned with various thresholds for controlling the trade-off between size and quality of the resulting representative set. The experiments demonstrated how good our method is for extracting in an readily interpretable way the main features from sets of sequences. The proposed tools are made available as functions of the TraMineR R-package for categorical sequence analysis but are indeed not limited to sequence data sets and can be applied to dissimilarity matrices representing distances between any object type.

## References

1. Abbott, A., Tsay, A.: Sequence analysis and optimal matching methods in sociology, Review and prospect. *Sociological Methods and Research* 29(1), 3–33 (2000) (With discussion, pp. 34–76)
2. Müller, N.S., Gabadinho, A., Ritschard, G., Studer, M.: Extracting knowledge from life courses: Clustering and visualization. In: Song, I.-Y., Eder, J., Nguyen, T.M. (eds.) DaWaK 2008. LNCS, vol. 5182, pp. 176–185. Springer, Heidelberg (2008)
3. Hobohm, U., Scharf, M., Schneider, R., Sander, C.: Selection of representative protein data sets. *Protein Sci.* 1(3), 409–417 (1992)
4. Holm, L., Sander, C.: Removing near-neighbour redundancy from large protein sequence collections. *Bioinformatics* 14(5), 423–429 (1998)
5. Gabadinho, A., Ritschard, G., Studer, M., Müller, N.: Mining sequence data in R with the TraMineR package: A user's guide. Technical report, Department of Econometrics and Laboratory of Demography, University of Geneva, Geneva (2009)
6. McVicar, D., Anyadike-Danes, M.: Predicting successful and unsuccessful transitions from school to work by using sequence methods. *Journal of the Royal Statistical Society. Series A (Statistics in Society)* 165(2), 317–334 (2002)
7. Kaufman, L., Rousseeuw, P.J.: Finding groups in data: an introduction to cluster analysis. John Wiley and Sons, New York (1990)
8. Studer, M., Ritschard, G., Gabadinho, A., Müller, N.S.: Discrepancy analysis of complex objects using dissimilarities. In: Guillet, F., Ritschard, G., Zighed, D.A., Briand, H. (eds.) *Advances in Knowledge Discovery and Management*. SCI, vol. 292, pp. 3–19. Springer, Heidelberg (2010)
9. Clark, R.D.: Optimis: An extended dissimilarity selection method for finding diverse representative subsets. *Journal of Chemical Information and Computer Sciences* 37(6), 1181–1188 (1997)
10. Daszykowski, M., Walczak, B., Massart, D.L.: Representative subset selection. *Analytica Chimica Acta* 468(1), 91–103 (2002)

# Unsupervised Quadratic Discriminant Embeddings Using Gaussian Mixture Models

Eniko Szekely, Eric Bruno, and Stephane Marchand-Maillet

University of Geneva, Viper Group

Battelle A, 7 Route de Drize, 1227 Geneva, Switzerland

{eniko.szekely, eric.bruno, stephane.marchand-maillet}@unige.ch

**Abstract.** We address in this paper the problem of finding low-dimensional representation spaces for clustered high-dimensional data. The new embedding space proposed here, called the *cluster space*, is an unsupervised dimension reduction method that relies on the estimation of a Gaussian Mixture Model (GMM) parameters. This allows to capture information not only among data points, but also among clusters in the same embedding space. Points are represented in the cluster space by means of their a posteriori probability values estimated using the GMMs. We show the relationship between the cluster space and the Quadratic Discriminant Analysis (QDA), thus emphasizing the discriminant capability of the representation space proposed. The estimation of the parameters of the GMM in high dimensions is further discussed. Experiments on both artificial and real data illustrate the discriminative power of the cluster space compared with other known state-of-the-art embedding methods.

## 1 Introduction

When mining data, detection of relevant information becomes important for knowledge discovery. Clusters and structures constitute such type of knowledge as in many applications, data is naturally organised into clusters. Moreover, the dimensionality of the data available today is increasing rapidly. In this context, an increasing interest has emerged in finding low-dimensional representation spaces for high-dimensional data. Dimension reduction (low-dimensional embedding) was motivated by: 1) the knowledge that data often lies in spaces of lower dimensionality than the original spaces; 2) the necessity to visualise data and 3) the need to reduce the computational load of high-dimensional data processing. Whereas existing embedding methods concentrate on revealing manifolds from within the data, we consider here the context of clustered data. The preservation of cluster information in the process of dimension reduction, despite its importance in numerous fields, has still received only little attention. Taking as an example the field of information retrieval and being given a document as a query, other documents' relevance to that query can be associated with the cluster membership (a document is more relevant to a query if it belongs to the same cluster as the query). In such a context, when reducing the dimensionality of the data, cluster preservation becomes important for efficient retrieval.

## 1.1 Related Work

Many different unsupervised approaches were proposed for the embedding of high-dimensional data into lower dimensional spaces. Principal Components Analysis is the most commonly used dimension reduction method. It tries to linearly capture as much as possible from the variance in the data, but it is not designed to cope with non-Gaussian distributions and even less with clustered data (mixture models). Embedding methods based on pairwise distances, like Multidimensional Scaling (MDS) [2], try to preserve as faithfully as possible the original Euclidean pairwise distances. Sammon Mapping in [15] was designed to increase the importance given to small distances in the minimization of the embedding function. Methods like Nonlinear MDS [2] try to preserve non-linear transformations of distances or to unfold data that lies on manifolds (Isomap [16], Curvilinear Component Analysis (CCA) [3], Curvilinear Distance Analysis (CDA) [12]).

Manifolds are non-linear structures where two points, even if close with respect to the Euclidean distance, can still be located far away on the manifold. Isomap and CDA use the *geodesic* distance, that is, the distance over the manifold and not through the manifold. Both CCA and CDA weight the distances in the output space and not in the input space like MDS, Isomap or Sammon Mapping do. Contrary to Isomap, which is a global method, Locally Linear Embedding [14] is a local method which tries to preserve the local structure - the linear reconstruction of a point from its neighbours. Similar to LLE, Laplacian Eigenmaps [1] build a neighbourhood graph and embed points with respect to the eigenvectors of the Laplacian matrix. Stochastic Neighbour Embedding [8], rather than preserving distances, preserves probabilities of points of being neighbours of other points. Pairwise distance-based methods were not initially designed to project new testing points in the reduced space, since the embedding had to be recomputed each time a new point was added. Further work for out-of-sample extensions was proposed to deal with this problem. The methods presented above rely on pairwise information, either distances or probabilities, and aim at recovering the hidden low-dimensional manifold on which the high-dimensional data lies, rather than preserving cluster information.

Clustering is generally approached through either hierarchical or partitional methods. Hierarchical algorithms output a hierarchy of clusters, either in a top-down or a bottom-up approach (divise vs. agglomerative). The hierarchy allows an inspection of the dataset at different levels. On the other side, partitional methods partition the data into different clusters by doing either a hard or a soft assignment. Hard clustering methods assign each point to exactly one cluster, creating disjoint clusters. The typical example is the  $k$ -means algorithm [13]. In soft clustering each point is assigned a different degree of belonging to each of the clusters, such as in the fuzzy clustering. Probabilistic Gaussian Mixture Model [7] assumes a mixture model of multiple gaussians and remains the most employed soft clustering method in practice.

The preservation of cluster information is precisely the aim of the cluster space proposed in this paper. The idea of projecting points in the space of the clusters has already

been approached in the literature. In [6] the authors propose a cluster space model in order to analyse the similarity between a customer and a cluster in the transactional application area. Their solution makes use of multiple hard clustering outputs and maps the results of the different clusterings into a common space, the cluster space. Analysis is further performed in this space in order to model the dynamics of the customers. The Parametric Embedding (PE) proposed in [9] embeds the posterior probabilities of points to belong to clusters in a lower-dimensional space using the Kullback-Leibler divergence. In PE, the posterior probabilities – result of a previous soft clustering step – are considered to be given as input to the algorithm. The same authors present in [10] the Probabilistic Latent Semantic Visualisation (PLSV) based on a topic model which assumes that both data points (documents) and clusters (topics) have latent coordinates in the low-dimensional visualisation space. In our approach we rely on the soft clustering results of a GMM model and use the log-scaling of the posterior probabilities, rather than the direct probabilities. We show that, this way, through the intrinsic relationship with QDA, we capture the discriminant information.

Supervised methods, that explore the class a priori information, like Linear Discriminant Analysis (LDA) [7] and Quadratic Discriminant Analysis (QDA) [7] are powerful tools used for efficient discrimination. However, in the current work, we are interested in unsupervised approaches for clustering and discrimination, where no class information is known in advance. We will show in the following, in the derivation of the cluster space, its intrinsic relationship to QDA, which gives it its discriminative power, even if unsupervised. Clustering and dimension reduction are powerful tools for the analysis of high-dimensional data. Combining these tools into a unified framework is a challenging task and represents the purpose of the *cluster space*. As shown later in the paper this allows for improved analysis and visualisation.

## 1.2 Motivation and Contributions of the Paper

Dimension reduction methods are often blind to the cluster structures present in high dimensions, making identification of clusters in reduced spaces difficult. Nevertheless, the need for structure preservation in the embedding process is important because, apart from a continuous inspection of the reduced space, many real-world applications rely on the recovery of structures from the original data.

In this paper we search for an embedding space  $\mathcal{S}$  – the *cluster space* – able to preserve and discriminate the clusters in the low-dimensional representation. In the definition of the *cluster space*, point coordinates are estimated by means of their relative distances to each one of the clusters. Thus, both interpoint and point-cluster information are captured and represented in the same embedding space. We assume that data comes from a mixture of Gaussians and will therefore model it using Gaussian Mixture Models. Once the cluster information is collected in the original space using GMM, the posterior probabilities provide the point coordinates in the cluster space. Considering that the estimation of the GMM parameters is optimal, the cluster space represents the optimal space for discrimination in terms of QDA.

The next section formally defines the *cluster space*. Experiments on artificial and real data and comparisons with other embedding methods are presented in Section 3. The paper ends with discussions and conclusions in Section 4.

## 2 The Cluster Space

Consider a set of  $N$  data points  $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$  that are assumed to come from a mixture model of  $K$  components. Data points originate from a  $D$ -dimensional space  $\mathbb{R}^D$ ,  $\mathbf{x}_i = \{x_i^1, x_i^2, \dots, x_i^D\}$  for all  $i = 1..N$ . Generally, the number of clusters ( $K$ ) is much smaller than the number of original dimensions ( $D$ ).

Multimodal distributions are often modelled by parametric mixture models, comprising of a number of mixtures, usually Gaussian. A Gaussian mixture model is a weighted sum of  $K$  Gaussian components parameterized by the mixture weights  $\pi_k$ ,  $k = 1..K$  and by the parameters of the Gaussians, the means  $\boldsymbol{\mu}_k$  and the covariances  $\boldsymbol{\Sigma}_k$ :

$$p(\mathbf{x}_i) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) . \quad (1)$$

where the weights satisfy the constraint:  $\sum_{k=1}^K \pi_k = 1$ .

The posterior probabilities of the mixtures in the GMM take the form:

$$p(k|\mathbf{x}_i) = \frac{\pi_k \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} . \quad (2)$$

To preserve the mixture (cluster) information intrinsic to (2), we define the *cluster space* as in the following.

**Definition 1.** *The cluster space  $\mathcal{S} = \{y_i^k\}$  is the low-dimensional embedding space of dimensionality  $K$  whose point coordinates  $y_i^k$  are estimated applying the logarithm to the posterior probabilities obtained from a Gaussian Mixture Model:*

$$y_i^k = -\log p(k|\mathbf{x}_i) . \quad (3)$$

The dimensionality of the *cluster space* is given by the number of clusters  $K$  and each point  $\mathbf{y}_i$  is represented by the  $K$  coordinates  $y_i^k$  – the coordinate of point  $\mathbf{y}_i$  along the dimension  $k$ .

With the *cluster space* defined, we shall take a closer look at the derivation of the coordinates in (3) to show the intrinsic relationship with the supervised Quadratic Discriminant Analysis.

### 2.1 Derivation of the Cluster Space

Applying the logarithm to the posterior probabilities from (2) we obtain the following coordinates  $y_i^k$  in the *cluster space*:

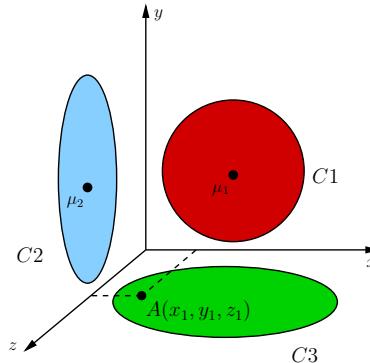
$$\begin{aligned}
y_i^k &= -\log \frac{\pi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \\
&= \frac{1}{2} \underbrace{\frac{1}{D_M^2} (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}_{+ \frac{1}{2} \log |\boldsymbol{\Sigma}_k| - \log \pi_k} + \frac{1}{2} \log(2\pi) \\
&\quad + \underbrace{\log \sum_j \pi_j \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}_{p(\mathbf{x}_i)}.
\end{aligned} \tag{4}$$

The log-scaling of the probabilities that appears in the equations of the coordinates  $y_i^k$  captures the Mahalanobis distance  $D_M$  from each point to each cluster center as indicated in (4). The Mahalanobis distance from a point to a cluster is the distance of that point to the center of the cluster divided by the width of the ellipsoid along the direction of the point. Thus, the Mahalanobis distance takes into account the shapes of the clusters expressed through the covariance matrices  $\boldsymbol{\Sigma}_k$ . A point close to a cluster in the Euclidean sense may be very far away in the Mahalanobian sense. It is therefore well suited for the *cluster space* as it allows to capture the cluster information contained not only in the interdistances between clusters but also in their shapes (Figure 1).

In equation (3) the minus sign allows the coordinates  $y_i^k$  to be positive (because the probabilities always take values in the interval [0,1], the log-scaling of probabilities is always negative). When estimating the coordinates in the cluster space, a particular case might appear, as illustrated in Figure 1 – that the probability that a point belongs to a cluster is zero ( $D_M = \infty$ ). The log-scaling of the probability becomes then  $-\infty$ . This case appears especially when working with high-dimensional data, where the subspaces spanned by different clusters are orthogonal (subspace clustering is an approach to clustering high-dimensional data that tries to find the subspace spanned by each of the clusters, see [11] for a review). We then replace the  $\infty$  values of the coordinates by a maximum value, thus pushing those points towards the boundary of the cluster space in the corresponding directions.

In equation (6) the constant  $\frac{D}{2} \log(2\pi)$  appears in the expression of the coordinates of every point with respect to each of the clusters, it therefore corresponds to a translation of the whole dataset with the same constant along each of the axis. The relative positions of each of the points with respect to all the other ones remain unchanged.

The cluster space can also be used as a gauge for clustering tendency since the more the clusters are separated, the larger the distances to the other clusters will be. Therefore the density of points around boundaries between clusters is a good indicator of class separability. A high density indicates a weak separation between the clusters, a low density indicating a high separability. Thus, further algorithms may be designed that use the *cluster space* as a mean for cluster tendency evaluation by analysing the distribution of points around boundaries.



**Fig. 1.** For each cluster  $C_1, C_2$  and  $C_3$  the subspace spanned has dimension 2. The Mahalanobis distance  $D_M$  from point  $A(x_1, y_1, z_1)$  to clusters  $C_1$  and  $C_2$  is Infinity. This information is captured in the cluster space.

## 2.2 Relationship with QDA

The Quadratic Discriminant Analysis is a supervised method for discrimination, whose quadratic discriminant functions (see [7] for further details) are given by:

$$\delta_k(\mathbf{x}_i) = \log \pi_k - \frac{1}{2} \log |\boldsymbol{\Sigma}_k| - \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) . \quad (5)$$

The coordinates  $y_i^k$  in the *cluster space* can be rewritten, with respect to the discriminant functions, as follows:

$$y_i^k = -\delta_k(\mathbf{x}_i) + \log p(\mathbf{x}_i) + \frac{D}{2} \log(2\pi) . \quad (6)$$

The presence of the discriminant functions in the derivation of the *cluster space* makes it the optimal space for discrimination in the framework of QDA given that the parameters of the GMM ( $\pi_k$ ,  $\boldsymbol{\mu}_k$  and  $\boldsymbol{\Sigma}_k$ ) are optimally estimated. This optimality depends on the choice of the number of clusters  $K$  (implicitly the dimensionality of the cluster space) and on the initialisation of the GMM model.

## 2.3 The Algorithm

The algorithm estimating the coordinates in the cluster space is summarised in Table 1. The inputs are the dataset  $\mathbf{X}$  and the number of clusters  $K$  and the outputs are the new coordinates in the cluster space  $\mathcal{S}$ . In Step 1 the priors  $\pi_k$ , means  $\boldsymbol{\mu}_k$  and covariances  $\boldsymbol{\Sigma}_k$  are estimated using the Expectation-Maximization (EM) algorithm [4]. The posterior probabilities  $p(k|\mathbf{x}_i)$  are estimated in Step 2. Finally, in Step 3, the point coordinates in the new space  $\mathcal{S}$  are estimated by applying the logarithm to the probabilities estimated in Step 2.

**Table 1.** The algorithm for the Cluster Space

<b>Input:</b>	$\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathbb{R}^D, i = 1..N$ $K$ -number of clusters
<b>Output:</b>	$\mathcal{S} = \{y_i^k\}, i = 1..N, k = 1..K$
<b>Step1:</b>	Estimate the parameters of the GMM model: priors $\pi_k$ , means $\boldsymbol{\mu}_k$ and covariances $\boldsymbol{\Sigma}_k$ .
<b>Step2:</b>	Compute the posterior probabilities: $p(k \mathbf{x}_i)$ .
<b>Step3:</b>	Compute the coordinates in the <i>cluster space</i> : $y_i^k = -\log p(k \mathbf{x}_i)$ .

## 2.4 High-Dimensional Data

In high-dimensional spaces, estimating all the parameters necessary for a full covariance model is difficult due to the sparsity of the data and the “curse of dimensionality”. Multiple solutions are possible:

1. One is provided by *parsimonious models*. Multiple parsimonious models have been proposed with varying complexities according to the specific models (intraclass and intercluster) of the covariance matrices chosen: full different covariances, full common covariance, spherical covariance (see [5] for a review). In the Full Gaussian Mixture Model (Full-GMM), the most general GMM model, each component is characterised by its own full covariance matrix  $\boldsymbol{\Sigma}_k$ .
2. A second solution is given by Truncated Singular Value Decomposition (T-SVD). Full-GMM are difficult to apply in high dimensions, as the covariance matrices are often ill-conditioned in high-dimensional spaces and the number of parameters to estimate is too large. In this case, an approximate estimation of the inverse of the covariance matrix can be resolved by using T-SVD. Let the covariance matrix  $\boldsymbol{\Sigma}$  be decomposed using SVD:

$$\boldsymbol{\Sigma} = UDU^T .$$

where  $D$  is a diagonal matrix containing the eigenvalues and  $U$  an orthogonal matrix of eigenvectors. The inverse then becomes:

$$\boldsymbol{\Sigma}^{-1} = U D^{-1} U^T .$$

It can be approximated with T-SVD using the first  $t$  eigenvectors and eigenvalues:

$$\boldsymbol{\Sigma}_t^{-1} = U_t D_t^{-1} U_t^T .$$

The Mahalanobis distance from each point to each cluster center can be rewritten:

$$\begin{aligned} D_M^2(\mathbf{x}_i, \boldsymbol{\mu}_k) &= (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \\ &= (\mathbf{x}_i - \boldsymbol{\mu}_k)^T U_{tk} D_{tk}^{-1} U_{tk}^T (\mathbf{x}_i - \boldsymbol{\mu}_k) \\ &= [U_{tk}(\mathbf{x}_i - \boldsymbol{\mu}_k)]^T D_{tk}^{-1} [U_{tk}(\mathbf{x}_i - \boldsymbol{\mu}_k)] . \end{aligned} \quad (7)$$

The above equation shows that the estimation of distances is always computed with respect to the space represented by the first  $t$  eigenvectors of each covariance matrix, which allows to capture the cluster's subspace information.

3. A third solution is possible by first denoising the high-dimensional data with a method such as PCA, and further start the analysis in this reduced space.

## 3 Experiments

### 3.1 Artificial Data

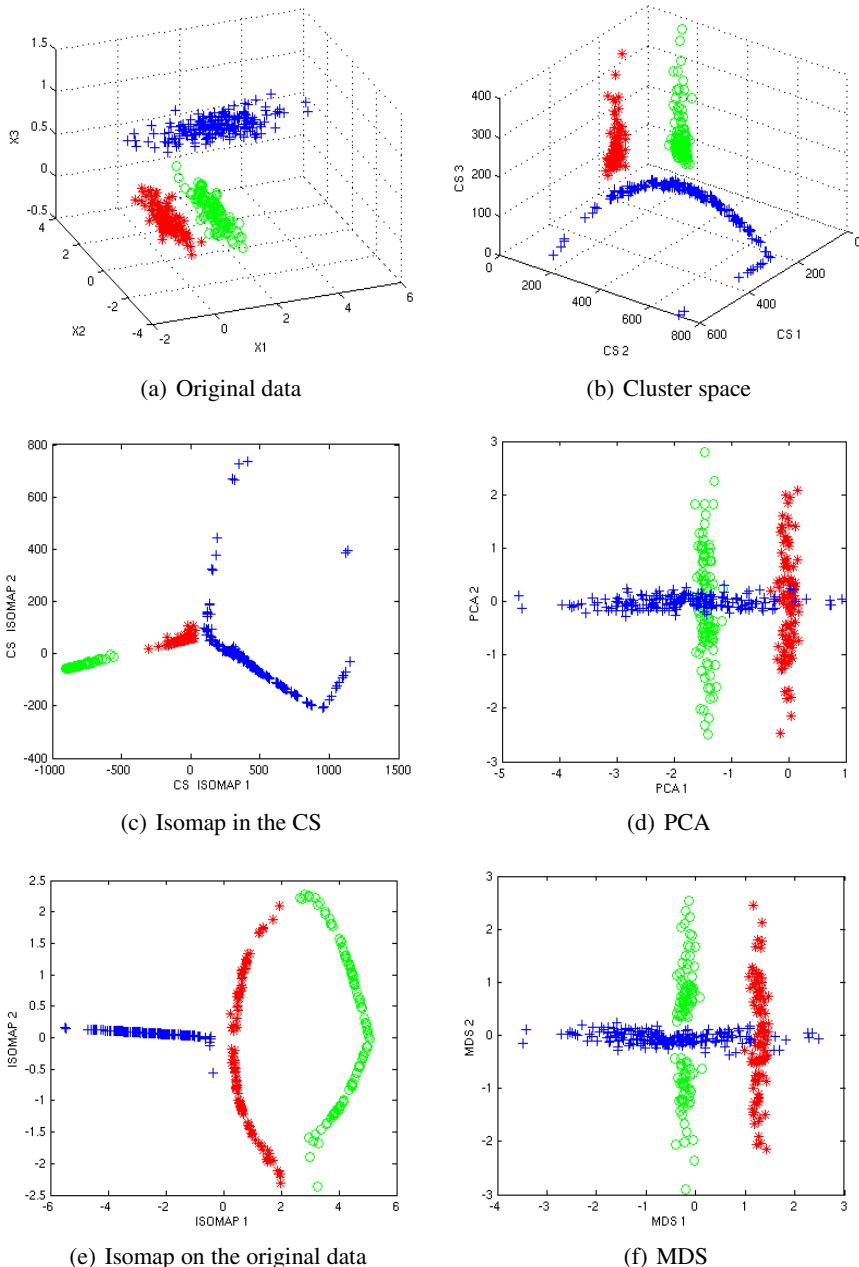
**Experiment 1.** We generate artificial data from 3 Gaussians each of 200 points in 3 dimensions as shown in Figure 2:  $\mathcal{N}_1(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ ,  $\mathcal{N}_2(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ ,  $\mathcal{N}_3(\boldsymbol{\mu}_3, \boldsymbol{\Sigma}_3)$  of 200 points with  $\boldsymbol{\mu}_1 = [0\ 0\ 0]$ ,  $\boldsymbol{\mu}_2 = [1.5\ 0\ 0]$ ,  $\boldsymbol{\mu}_3 = [1.5\ 0\ 1]$  and the covariances:

$$\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \begin{pmatrix} 0.01 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.01 \end{pmatrix}, \quad \boldsymbol{\Sigma}_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{pmatrix}. \quad (8)$$

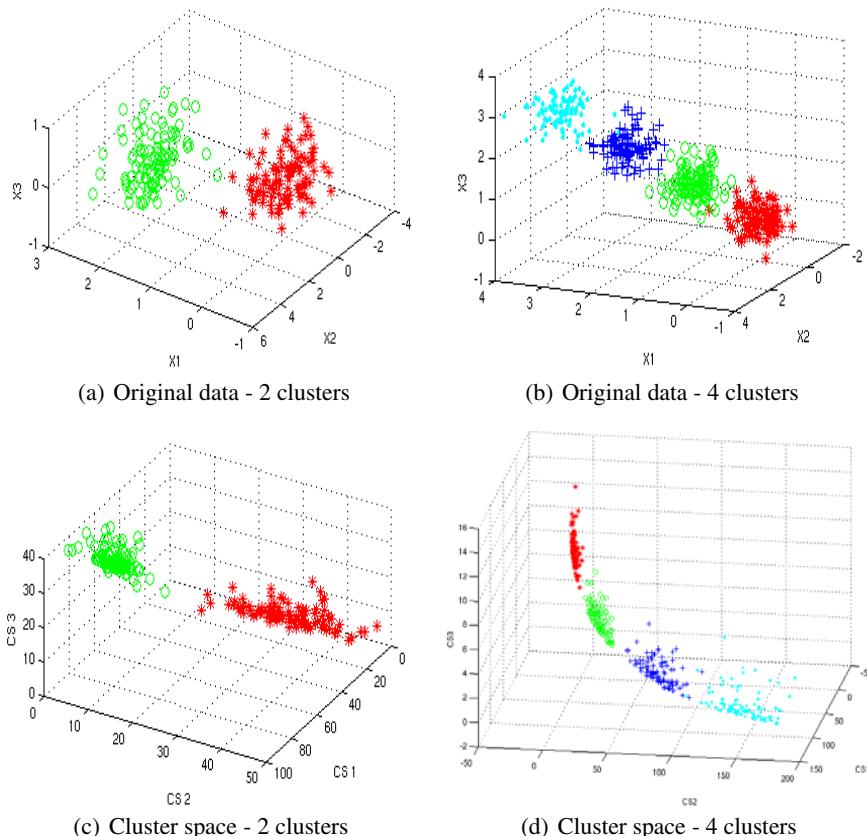
**Results.** We see in Figure 2 that algorithms like PCA (d) and MDS (f) are not capable of separating the 3 clusters that are well separated in (a). In the *cluster space* (b) the clusters are well separated. A further dimension reduction in this space using Isomap with a Manhattan distance shows in (c) that the clusters are separated. Isomap (e) also gives good results but is dependent on the number of neighbours given to build the fully connected graph (in such cases - of well separated clusters - the number of neighbours should be quite high).

**Experiment 2.** The choice of  $K$  (the number of clusters, and implicitly the dimension of the cluster space) plays an important role on the quality of the embedding in the cluster space. Figure 3 presents two cases when the number of chosen  $K$  is different from the number of real clusters in the data. We wish to show that the choice of  $K$  does not force unclustered data to be clustered and does not merge different clusters together.  $K$  is kept fixed ( $K = 3$ ) and the number of real clusters varies. We chose  $K = 3$  to be able to visualise the results in a 3D space.

**Results.** In the first example of Figure 3, (a) and (c),  $K$  is higher than the number of clusters and we observe that a higher  $K$  does not force clusters to break. This is an important aspect since the embedding, even if based on an initial clustering, should not artificially create structures that do not exist inside the data itself. Using a soft clustering like GMM avoids forcing clusters to break, like it would happen in a hard clustering approach ( $k$ -means). In the second example, (b) and (d),  $K$  is lower than the number of clusters and we observe that the algorithm does not merge different clusters. In conclusion, the choice of  $K$  is important but a number of situations work well even with different values. However, as observed during experimentation, a lower  $K$  influences more drastically the quality than a higher  $K$ , thus using higher estimates for  $K$  is preferred.



**Fig. 2.** Artificial data from 3 gaussians in 3 dimensions reduced using dimension reduction methods: a) Original data in 3 dimensions; b) Data projected in the *cluster space* using an EM with full covariances,  $K = 3$  and the Euclidean distance; c) Data from b) reduced using Isomap with the Manhattan distance and 30 neighbours to build the graph; d) PCA in the original space; e) Isomap in the original space with the Euclidean distance and 30 neighbours; f) MDS in the original space with the Euclidean distance.



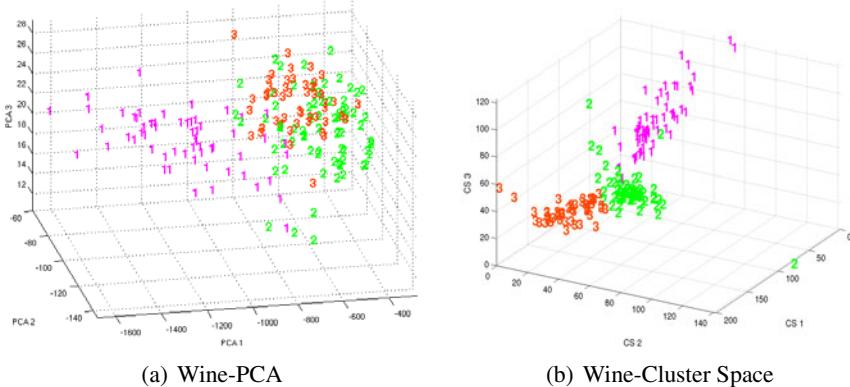
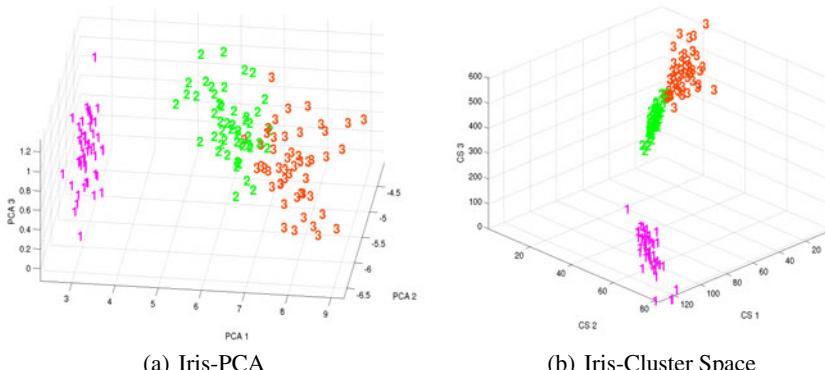
**Fig. 3.** The performance of the *cluster space* for cases when the assumed number of clusters  $K$  (here  $K = 3$ ) is different from the real number of clusters:  $K = 2$  (a, c) and  $K = 4$  (b, d)

### 3.2 Real-World Data

**Experiment 1.** We give a first example using two datasets from the UCI Machine Learning Repository: Wine and Iris. The Wine dataset contains 3 clusters with 178 data points in a 13-dimensional space. The embedding of the dataset in a 3-dimensional space is showed in Figure 4. We also evaluated the embeddings using Mean Average Precision (MAP),  $k$ -means purity and  $k$ NN ( $k$  Nearest Neighbour) error for  $k = 5$ . Mean average precision (MAP) is the mean of the average precisions over all data points and is a good indicator of the global quality of the embedding in the low-dimensional space.  $k$ -means purity estimates the accuracy of the  $k$ -means clustering in both the original and the low-dimensional spaces. The accuracy is estimated over three runs of  $k$ -means and the mean values are indicated. Results are presented in Table 2. We observe the high performance of the cluster space compared with the other methods tested for all the three evaluation methods.

**Table 2.** Evaluation of the Wine dataset

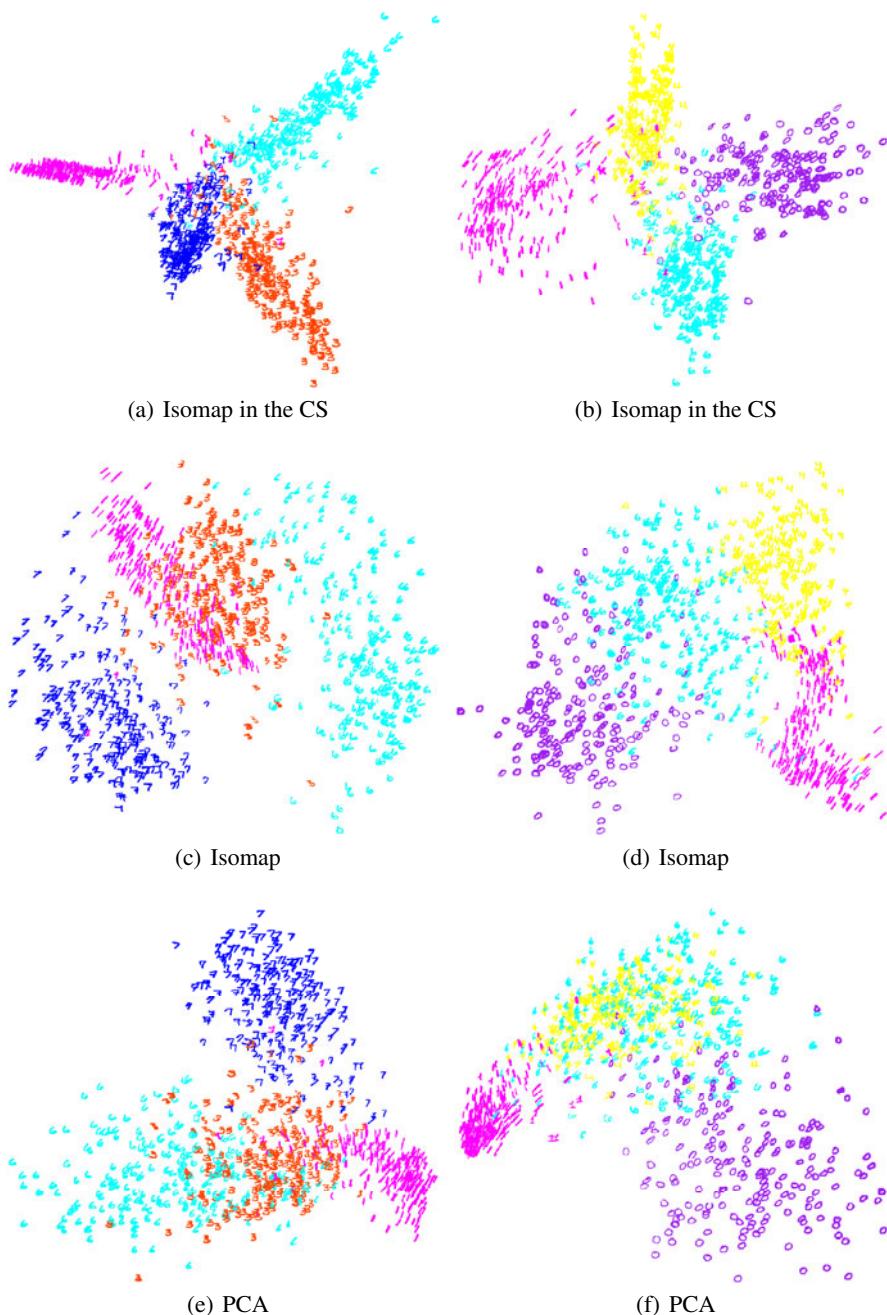
Wine	Orig	PCA	Sammon	Isomap	CS
Purity	67.42	70.22	69.29	69.28	84.83
MAP	0.6433	0.6422	0.6429	0.6424	0.8499
kNN	69.66	69.66	69.66	68.54	94.94

**Fig. 4.** Dimension reduction for the Wine dataset**Fig. 5.** Dimension reduction for the Iris dataset

The Iris dataset contains 150 data points grouped into 3 clusters and represented in a 4-dimensional feature space. Figure 5 shows the embeddings for PCA and the Cluster Space for the Iris dataset.

**Results.** For both the Wine and the Iris datasets, we observe that the cluster space better discriminates the different clusters as opposed to other dimension reduction methods like PCA, Sammon or Isomap.

**Experiment 2.** One of the main application of the cluster space can be seen as a preprocessing step for further data analysis. The cluster space is useful as a preprocessing



**Fig. 6.** Dimension reduction for 1000 MNIST digits (1, 3, 6, 7) (a, c, e) and 1000 MNIST digits (0, 1, 4, 6) (b, d, f)

step especially when a lower-dimensional space of dimension 2 or 3 is desired for example for visualisation. In this case, a second dimension reduction can be performed in the cluster space using a distance metric that preserves data geometry. To illustrate this, we use a high-dimensional dataset: the MNIST handwritten digits dataset, originally embedded in a 784-dimensional space.

**Results.** Two experiments are performed on two datasets, each of 1000 data points. The first dataset contains handwritten digits of 1, 3, 6 and 7 and the second dataset the handwritten digits of 0, 1, 4 and 6. The data points are embedded in the cluster space with dimensionality 4. Because of the high-dimensionality of the handwritten data, we applied Truncated SVD to estimate an approximation of the inverse of the covariance matrix. The first 70 eigenvectors of the covariance matrices were used for T-SVD. For visualisation, a second dimension reduction from the cluster space using Isomap was performed into a 2-dimensional space. The examples presented in Figure 6 show that the preprocessing in the cluster space helps Isomap to better separate clusters in the 2D space, compared to Isomap or PCA<sup>1</sup>.

## 4 Discussion and Conclusions

The current construction of the *cluster space* leads to the representation of the data in a lower-dimensional space that emphasizes clusters. The estimation of coordinates in this reduced space relies on the estimation of the parameters of a GMM model. The performance of GMM models is known to depend on the choice of the input parameters, that is the number of clusters  $K$  and the initialization of the cluster centers. The same parameters will influence the performance of the cluster space, whose dimensionality is, by construction, directed by the choice of the number of clusters. Results on artificial data (Figure 3) showed that this parameter is important, still various values provide good performances. However higher values of  $K$  are to be preferred.

One advantage of the presented model is the possibility of projection for new points in the *cluster space* (as long as they do not represent new clusters), their coordinates being computed from their distances relative to all of the clusters.

In conclusion, we presented a new representation space for embedding clustered data. Typically, the data is mapped onto the  $K$ -dimensional space, where  $K$  is given by the number of clusters and the coordinates are given by the log-scaling of the posterior probabilities estimated in an unsupervised manner usinng Gaussian Mixture Models. We call this reduced space the *cluster space*. This space is optimal for discrimination in terms of QDA when the parameters of the GMM are optimally estimated and is a good preprocessing step before applying other dimension reduction methods, such as Isomap.

**Acknowledgements.** This work has been partly funded by SNF fund No. 200020-121842 in parallel with the Swiss NCCR(IM)2.

---

<sup>1</sup> Our approach is unsupervised and, in all experiments, label information was used only for visualisation and evaluation.

## References

1. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Advances in Neural Information Processing Systems, vol. 14 (2002)
2. Borg, I., Groenen, P.: Modern multidimensional scaling: Theory and applications. Springer, Heidelberg (2005)
3. Demartines, P., Hérault, J.: Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. IEEE Transactions on Neural Network (1997)
4. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society 39 (1977)
5. Fraley, C., Raftery, A.: Model-based clustering, discriminant analysis and density estimation. Journal of American Statistical Association, 611–631 (2002)
6. Gupta, G., Ghosh, J.: Detecting seasonal trends and cluster motion visualization for very high-dimensional transactional data. In: Proceedings of the First International SIAM Conference on Data Mining (2001)
7. Hastie, T., Tibshirani, R., Friedman, J.: The elements of statistical learning. Springer, Heidelberg (2001)
8. Hinton, G., Roweis, S.: Stochastic neighbor embedding. In: Advances in Neural Information Processing Systems (2002)
9. Iwata, T., Saito, K., Ueda, N., Stromsten, S., Griffiths, T., Tenenbaum, J.: Parametric embedding for class visualization. Neural Computation (2007)
10. Iwata, T., Yamada, T., Ueda, N.: Probabilistic latent semantic visualization: topic model for visualizing documents. In: Proceedings of the 14th ACM SIGKDD, USA, pp. 363–371 (2008)
11. Kriegel, H.-P., Kroger, P., Zimek, A.: Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering and correlation clustering. ACM Transactions on Knowledge Discovery from Data (TKDD) 3 (2009)
12. Lee, J., Lendasse, A., Verleysen, M.: A robust nonlinear projection method. In: Proceedings of ESANN 2000, Belgium, pp. 13–20 (2000)
13. MacQueen, J.B.: Some Methods for Classification and Analysis of MultiVariate Observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297 (1967)
14. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. Science 290, 2323–2326 (2000)
15. Sammon, J.W.: A nonlinear mapping for data structure analysis. IEEE Transactions on Computers C-18 (1969)
16. Tenenbaum, J., de Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. Science 290, 2319–2323 (2000)

# How to Rank Terminology Extracted by EXTERLOG

Hassan Saneifar<sup>1,2</sup>, Stéphane Bonniol<sup>2</sup>, Anne Laurent<sup>1</sup>,  
Pascal Poncelet<sup>1</sup>, and Mathieu Roche<sup>1</sup>

<sup>1</sup> LIRMM - Univ. Montpellier 2 - CNRS, Montpellier, France

<sup>2</sup> Satin IP Technologies, Montpellier, France

{saneifar, laurent, poncelet, mroche}@lirmm.fr

<http://www.lirmm.fr/~saneifar,laurent,poncelet,mroche>

**Abstract.** In many application areas, systems reports occurring events in a kind of textual data called usually log files. Log files report the status of systems, products, or even causes of problems that can occur. The Information extracted from log files of computing systems can be considered one of the important resources of information systems. Log files are considered as a kind of “complex textual data”, i.e. *the multi-source, heterogeneous, and multi-format* data. In this paper, we aim particularly at exploring the lexical structure of these log files in order to extract the terms used in log files. These terms will be used in the building of domain ontology and also in enrichment of features of log files corpus. According to features of such textual data, applying the classical methods of information extraction is not an easy task, more particularly for terminology extraction. Here, we introduce a new developed version of EXTERLOG, our approach to extract the terminology from log files, which is guided by Web to evaluate the extracted terms. We score the extracted terms by a *Web* and *context* based measure. We favor the more relevant terms of domain and emphasize the precision by filtering terms based on their scores. The experiments show that EXTERLOG is well-adapted terminology extraction approach from log files.

**Keywords:** Natural language processing, Information retrieval, Terminology extraction, Terminology ranking, Log files.

## 1 Introduction

In many applications, computing systems generate reports automatically. These digital reports, also known as system logs, represent the major source of information on the status of systems, products, or even the causes of problems that can occur. Although log files are generated in every field of computing, the characteristics of these logs, particularly the language, structure and context, differ from system to system. In some areas, such as Integrated Circuit (IC) design systems, the log files are not systematically exploited in an effective way whereas in this particular field, the log files generated by IC design tools, contain essential information on the condition of production and the final products. In this context, a key challenge is to provide approaches which consider *the multi-source, heterogeneous and scalable structures* of log files as well as their *special vocabulary*. Furthermore, although the contents of these logs are similar to texts written in Natural Language (NL), they comply neither with the grammar nor with

the NL structure. Therefore, In order to extract information from the logs, we need to adapt Natural Language Processing (NLP) and Information Extraction (IE) techniques to the specific characteristics of such textual data. Another key challenge is evaluation of results. In fact, according to the particularity of such data, and then due to the high noise ratio in results, the classic evaluation methods are not necessarily relevant. To emphasize the precision of results as a must according to the accuracy of context, we have to define the noise filtering method which comply with the particularity of such data.

The creation of a domain ontology is a primordial need for our future work on information extraction from log files. Defining the vocabulary of domain is one of the first steps of building an ontology. To analyze vocabulary and lexical structure of a corpus, extraction of domain terminology is one of the most important phases. We thus aim at extracting the terminology of log files. The extracted terms will be used in the creation of domain ontology in our future works. Also, we will use extracted terms to study the different lexical structures of different logs in order to enrich our information extraction methods. In this paper, we introduce a new version of our approach EXTERLOG (EXtraction of TERminology from LOGs), previously presented in [1], that is developed to extract the terminology from these log files. In this approach, we study how to adapt the existing terminology extraction methods to the particular and heterogeneous features of log files. We also present in this paper a filtering method of extracted terms based on a ranking score in order to emphasize the precision of extracted relevant terms.

In Sect. 2, we detail the utility of building domain ontology and thus the terminology extraction in our context and the special features and difficulties of this domain. Our approach EXTERLOG is developed in Sect. 3. Section 4 describes and compares the various experiments that we performed to extract terms from the logs and specially to evaluate the precision of EXTERLOG.

## 2 Context

Today, digital systems generate many types of log files, which give essential information on the systems. Some types of log files, like network monitoring logs, web services interactions or web usage logs are widely exploited [2][3]. These kinds of log files are based on the management of events. That is, the computing system, which generates the log files, records the system events based on their occurring times. The contents of these logs comply with norms according to the nature of events and their global usage (*e.g.* web usage area).

However, in some areas such as integrated circuit design systems, rather than being some recorded events, the generated log files are digital reports on configuration, condition and states of systems. The aim of the exploitation of these log files is not to analyze the events but to extract information about system configuration and especially about the final product's condition. Hence, log files are considered an important source of information for systems designed to query and manage the production. Information extraction in log files generated by IC design tools has an attractive interest for automatic management and monitoring of IC production. However, several aspects of these log files have been less emphasized in existing methods of text mining and NLP. These specific characteristics raise several challenges that require more research.

## 2.1 IE and Log Files

To use these logs in an information system, we must implement information extraction methods which are adapted to the characteristics of these logs. Moreover, these features explain why we need a domain ontology to extract information from the log files.

In the field of integrated circuits design, several levels need to be considered. At every level, different design tools can be used which make the generated log files the *multi-source* data. Despite the fact that the logs of the same design level report the same information, their structures can differ significantly depending on the design tool used. Specifically, each design tool often uses its own vocabulary to report the same information. In the verification level, for example, we produce two log files (*e.g.* log “A” and log “B”) by two different tools. The information about, for example, the “Statement coverage” will be expressed as follows in the log “A”:

TOTAL	COVERED	PERCENT
Lines	10	11
statements	20	21

But the same information in the log “B”, will be disclosed from this single line:

EC: 2.1%

As shown above, the same information in two log files produced by two different tools is represented by different structures and vocabulary. Moreover, design tools evolve over time and this evolution often occurs unexpectedly. Therefore, the *format of the data* in the log files changes, which make the automatic management of data difficult. The *heterogeneity* of data exists not only between the log files produced by different tools, but also within a given log file. For example, the symbols used to present an object, such as the header for tables, change in a given log. Similarly, there are several formats for punctuation, the separation lines, and representation of missing data. Therefore, we need intelligent and generalized methods, which can be applied at the same time on different logs (*multi-source textual data*) which have the multi-format and heterogeneous data. These methods must also take into account the variable vocabulary of these logs. To generalize the extraction methods, we thus need to identify the terms used by each tool in order to create the domain ontology. This ontology allows us to better identify equivalent terms in the logs generated by different tools and so to reduce the heterogeneity of data. For instance, to check “Absence of Attributes” as a query on the logs, one must search for the following different sentences in the logs, depending on the version and type of design tool used:

- "Do not use **map\_to\_module attribute**"
- "Do not use **one\_cold or one\_hot attributes**"
- "Do not use **enum\_encoding attribute**"

Instead of using several patterns, each one adapted for a specific sentence, by associating the words “*map\_to\_module attribute*”, “*one\_hot attributes*” and “*enum\_encoding attribute*” to the concept “*Absence of Attributes*”, we use a general pattern that expands automatically according to different logs using

the domain ontology. The ontology-driven expansion of query is studied in many works, see [4][5].

The ontology will allow us to better identify equivalent terms in the logs generated by different tools. Several approaches are based on the domain ontology to better guide the information extraction [6]. An ontology also defines the common vocabulary of a domain [7]. In our context, the domain ontology allows us to categorize the terms associated with a concept sought on the logs. The creation of ontology requires a lexical analysis of a corpus to identify the terms of the domain. We hence seek to identify the terms of the logs of every design tool. We will then look at these terms in order to make the correspondence between them and to create the domain ontology. Thus, we aim at studying the extraction of terminology from log files.

Also, the language used in these logs is a difficulty that affects the methods of information extraction. Although the language used in these logs is English, the contents of these logs do not usually comply with “classic” grammar. Moreover, there are words that are often constituted from alphanumeric and special characters.

Due to these specific characteristics of log files, the methods of NLP, including the terminology extraction, developed for texts written in natural language, are not necessarily well suited to the log files.

## 2.2 Terminology Extraction Background

The extraction of domain terminology from the textual data is an essential task to establish specialized dictionary of a domain [8]. The extraction of co-occurring words is an important step in identifying the terms. To identify the co-occurrences, some approaches are based on syntactic techniques which rely initially on the grammatical tagging of words. The terminological candidates are then extracted using syntactic patterns (*e.g.* adjective-noun, noun-noun). We develop the grammatical tagging of log files using our approach EXTERLOG in Sect. 3.2.

Bigrams<sup>1</sup> are used in [9] as features to improve the performance of the text classification. The series of three words (*i.e.* trigrams) or more is not always essential [10]. The defined rules and grammar are used in [11] in order to extract the nominal terms as well as to evaluate them. The machine learning methods based on Hidden Markov Models (HMMs) are used in [12] to extract terminology in the field of molecular biology. EXIT, introduced by [8] is an iterative approach that finds the terms in an incremental way. A term found in an iteration is used in the next one to find more complex terms. Some works try to extract the co-occurrences in a fixed size window (*normally five words*). In this case, the extracted words may not be directly related [13]. XTRACT avoids this problem by considering the relative positions of co-occurrences. XTRACT is a terminology extraction system, which identifies lexical relations in the large corpus of English texts [14]. SYNTEx, proposed by [15], performs syntactic analysis of texts to identify the names, verbs, adjectives, adverbs, the noun phrases and verbal phrases. It analyses the text by applying syntactic rules to extract terms.

As described above, we have previously studied the extraction of terminology based on identifying the co-occurring words *without* using the syntactic patterns from log files

---

<sup>1</sup> N-grams are defined as the series of any “n” words.

(see [1]). As explained in [1], the terminology extraction based on syntactic patterns is quite relevant to the context of log files. We shown that the accuracy of terms extracted based on syntactic patterns is indeed higher than the precision of bigrams extracted without such patterns. Despite the fact that normalization and tagging the texts of logs is not an easy task, our previous experiments show that an effort in this direction is useful in order to extract quality terms. But according to the need of high accuracy in this domain and the fact that manual validation of terms by an expert is expensive, we develop here the automatic evaluation phase of EXTERLOG. This evaluation of terms is detailed in Section 3.4.

The statistical methods used are generally associated with syntactic methods for evaluating the adequacy of terminological candidates [16]. These methods are based on statistical measures such as information gain to validate an extracted candidate as a term. Among these measures, the occurrence frequency of candidates is a basic notion. However, these statistical methods are not relevant to be applied on the log files. Indeed, statistical approaches can cope with high frequency terms but tend to miss low frequency ones [17]. According to the log files described above, the repetition of words is rare. Each part of a log file contains some information independent from other parts. In addition, it is not reasonable to establish a large corpus of logs by gathering log files generated by the same tool at the same level of design. Since, it just results the redundancy of words. Evaluation of terms based on some other resources like as web is studied by many works. The Web, as a huge corpus, is more and more used in NLP methods specially in validation of results. However, in our context, we study the corpus of a very specialized domain. The terms used in this domain are the specialized terms and not frequently seen on the Web. Then, we could not use the classic statistical measures based on simple frequencies of terms in corpus in order to give a score to every extracted term. Furthermore, our approach aims at reducing the noise ratio in results, thus emphasizing the precision, by filtering the extracted terms using a web based statistical measures which considers in the same time the context of log files. We detail this aspect in Sect. 3.4.

A lot of works compare the different techniques of terminology extraction and their performance. But most of these studies are experimented on textual data, which are classical texts written in natural language. Most of the corpus that are used are structured in a consistent way. In particular, this textual data complies with the grammar of NL. However, in our context, the characteristics of logs (such as not to comply with natural language grammar, their heterogeneous and evolving structures (cf. Sect. 2)) impose an adaptation of these methods to ensure that they are relevant for the case of log files.

### **3 EXTERLOG: EXtraction of TERminology from LOGs**

Our approach, EXTERLOG, is developed to extract the terminology in the log files. The extraction process involves normalization, preprocessing of log files and grammatical tagging of word in order to extract the terms. EXTERLOG contains also a filtering phase of extracted terms based on a scoring measure.

### 3.1 Preprocessing and Normalization

The heterogeneity of the log files is a problem, which can affect the performance of information extraction methods. In order to reduce the heterogeneity of data and prepare them to extract terminology, we apply a series of preprocessing and normalization on the logs. Given the specificity of our data, the normalization method, adapted to the logs, makes the format and structure of logs more consistent. We replace the punctuations, separation lines and the headers of the tables by special characters to limit ambiguity. Then, we tokenize the texts of logs, considering that certain words or structures do not have to be tokenized. For example, the technical word “Circuit4-LED3” is a single word which should not be tokenized into two words “Circuit4” and “LED3”. Besides, we distinguish automatically the lines representing the header of tables from the lines which separate the parts. After the normalization of logs, we have less ambiguity and less common symbols for different concepts. This normalization makes the structure of logs produced by different tools more homogeneous.

### 3.2 Grammatical Tagging

Grammatical tagging (also called *part-of-speech tagging*) is a method of NLP used to analyse the text files which aims to annotate words based on their grammatical roles. In the context of log files, there are some difficulties and limitations for applying a grammatical tagging on such textual data.

Indeed, the classic techniques of POS tagging are developed using the standard grammar of natural language. In addition, they are normally trained on texts written in a standard natural language, such as journals. Therefore, they consider that a sentence ends with a fullstop, for example, which is not the case in the log files that we handle. More specifically, in these log files, sentences and paragraphs are not always well structured. Besides, there are several constructions that do not comply with the structure of sentences in natural language. To identify the role of words in the log files, we use BRILL rule-based part-of-speech tagging method [18]. Since existing taggers like BRILL are trained on general language corpora, they give inconsistent results on the specialized texts. [19] propose a semi-automatic approach for tagging corpora of speciality. They build a new tagger which corrects the base of rules obtained by BRILL tagger and adapt it to a corpus of speciality. In the context of log files, we need also to adapt BRILL tagger just as in [19]. We thus adapted BRILL to the context of log files by introducing the new *contextual* and *lexical* rules. Since, the classic rules of BRILL, which are defined according to the NL grammar, are not relevant to log files. For example, a word beginning with a number is considered a “*cardinal*” by BRILL. However, in the log files, there are many words like 12.1vSo10 that must not be labelled as “*cardinal*”. Therefore, we defined the special *lexical* and *contextual* rules in BRILL. The structures of log files can contribute important information for extracting the relevant patterns in future works. Therefore, we preserve the structure of files during grammatical tagging. We introduce the new tags, called “*Document Structure Tags*”, which present the different structures in log files. For example, the tag “\TH” represents the header of tables or “\SPL” represents the lines separating the log parts. The special structures in log files are identified during normalization by defined rules. Then, they are identified during tagging by the new specific contextual rules defined in BRILL. We finally get the

logs tagged by the grammatical roles of words and also by the labels that determine the structure of logs.

### 3.3 Extraction of Co-occurrences

We extract the co-occurrences in the log files respecting a defined *part-of-speech* syntactic pattern. We call the co-occurrences extracted using syntactic pattern “POS-candidates”<sup>2</sup>. The syntactic patterns determine the adjacent words with the defined grammatical roles. The syntactic patterns are used in [16] and [15] to extract terminology. As argued in [16], the base structures of syntactic patterns are not frozen structures and accept variations. According to the terms found in our context, the syntactic patterns that we use to extract the “POS-candidates” from log files are:

- “\JJ - \NN” (Adjective-Noun),
- “\NN - \NN” (Noun-Noun).

These extracted terms at this phase must be scored to favor the most relevant terms of the domain.

### 3.4 Filtering of Candidates

All the extracted terms are not necessarily the relevant terms of the domain. Because of some huge log files and the large vocabulary of the logs, there exists so many extracted terms. Also, according to the particular features of such data, in spite of adapted normalization and tagging methods that we used, there exists some noise (no relevant terms) in the extracted terms. Moreover, we are focused on a specialized domain where just some terms are really bidden to the domain’s context. Thus, we score, rank and then filter the extracted terms in order to favor the most relevant terms according to the context. The statistical measures are often used in terminology extraction field to evaluate the terms (see [20]). The following ones are the most widely used.

**Mutual Information.** One of the most commonly used measures to compute a sort of relationship between the words composing what is called a **co-occurrence** is Church’s Mutual Information (MI) [21]. The simplified formula is the following where  $nb$  designates the number of occurrences of words and couples of words:

$$MI(x, y) = \log_2 \frac{nb(x, y)}{nb(x)nb(y)}$$

**Cubic Mutual Information.** The Cubic Mutual Information is an empirical measure based on MI, that enhances the impact of frequent co-occurrences, something which is absent in the original MI [22].

$$MI3(x, y) = \log_2 \frac{nb(x, y)^3}{nb(x)nb(y)}$$

This measure is used in several works related to noun or verb terms extraction in texts [23].

---

<sup>2</sup> POS: Part-Of-Speech.

**Dice's Coefficient.** An interesting quality measure is Dice's coefficient [24]. It is defined by the following formula based on the frequency of occurrence.

$$Dice(x, y) = \frac{2 \times nb(x, y)}{nb(x) + nb(y)}$$

These measures are based on the occurrence frequencies of terms in corpus. Scoring the terms based on frequencies of terms in corpus of logs is not a relevant approach in our context. As we have already explained, the techniques based on frequency of terms in a corpus (e.g. pruning terms having low frequency) are not relevant to this context as a *representative term* does *not* necessarily have a *high frequency* in log files. That is why we score the terms according to their frequencies on the Web as a large corpus where frequency of a term can be representative. Working on a specialized domain, we have bias scores based on the simple count of occurrences of a term on Web. Indeed, on Web, we capture occurrences of terms regardless of the context in which they are seen. Thus, we should consider only the occurrences of terms on web which are situated in the IC design context. We use therefore an extension of described measures called *AcroDef*. *AcroDef* is a quality measure where context and Web resources are essential characteristics to be taken into account (see [23]). The below formulas define the *AcroDef* measures, respectively based on MI and Cubic MI.

$$AcroDef_{MI}(a^j) = \frac{nb(\bigcap_{i=1}^n a_i^j + C)}{\prod_{i=1}^n nb(a_i^j + C | a_i^j \notin M_{stop-words})}$$

where  $n \geq 2$

$$AcroDef_{MI3}(a^j) = \frac{nb(\bigcap_{i=1}^n a_i^j + C)^3}{\prod_{i=1}^n nb(a_i^j + C | a_i^j \notin M_{stop-words})}$$

where  $n \geq 2$

In *AcroDef*, the **context** “*C*” is represented as a set of significant words. The *nb* function used in the preceding measures represents the number of pages provided by the search engine to given query. Then  $nb(a_i^j + C)$  returns the number of pages applying query  $a_i^j + C$  which means all words of the term  $a^j$  in addition to those of context *C*. In our case, for example, for a term  $x^j$  like “atpg patterns” consisting of two words (so  $i = 2$ ),  $nb(atpg \cap patterns + C)$  is the number page returned by applying query “atpg pattern” AND *C* on a search engine, where *C* is a set of words representing the context. The *AcroDefDice* formula based Dice's formula is written as follows:

$$\frac{|\{a_i^j + C | a_i^j \notin M_{stop-words}\}_{i \in [1, n]}| \times nb(\bigcap_{i=1}^n a_i^j + C)}{\sum_{i=1}^n nb(a_i^j + C | a_i^j \notin M_{stop-words})}$$

where  $n \geq 2$

In [23], “*C*” is represented as a set of significant words (e.g. encryption, information and code to represent the Cryptography context). The authors made some experiments with different number of words represented as context. In all cases, authors use “AND” search engine operator between the words of context. That is, they request the pages containing all words in “*C*”. However, working on a very specialized domain which

contains some more specific sub domains, we do not get the best results by using just an “AND” operator for the words of context.

To specify the words which represent the context of log files, we build a corpus of documents including the reference documents of Integrated Circuit design tools and three other domains documents. We rank the words of corpus by using tf-idf measure (see [25]). Tf-idf gives higher score to the frequent words of a domain which are not frequent in other ones. Then, we choose the first five words (ranked in tf-idf order) of IC design documents as representing word of the context. As argued above, we look for web pages containing a given term and *two or more* words of context (*using the operators OR and AND*). Finally, the extracted terms are ranked by means of their *AcroDef* scores. We favor the most ranked terms by filtering those having most low *AcroDef* scores.

## 4 Experiments

In all experiments the log corpus is composed of logs of five levels of IC design. For each level, we considered two logs generated in different conditions of design systems. The size of the log corpus is about 950 KB. All the experiments are done using the Google search engine. Here, we are looking to study *AcroDef* ranking function and its ability to give a high score to relevant terms and low score to no relevant ones. The extracted terms by EXTERLOG from the log files are so numerous which make difficult the final validation by experts of domain. Thus, we experiment by taking a sample of extracted terms. We select the 200 more frequent terms extracted from logs of every IC design level. Note that in few levels, there exists less than 200 terms. The taken sample consists of 700 terms at all.

To filter the extracted terms from log files, we rank them by *AcroDef* (cf. 3.4). To apply *AcroDef*, we determine the context words as described in Sect. 3.4. Then, we use the Google search engine to capture the number of pages containing a given term and *two or more* words of context. Suppose a given term like “CPU time” where  $C_i$   $i \in \{1 - 5\}$  are the context words, the query used in Google search engine is “CPU time” AND  $C_1$  AND  $C_2$  OR  $C_3$  OR  $C_4$  OR  $C_5$ .

Once *AcroDef* scores are calculated, we rank the terms based on their *AcroDef*. The more *AcroDef* has a higher value, the more the term is representative (*seen*) in our context. Then, we select the most rated terms in the goal of emphasizing the precision by reducing the noise ratio (no relevant terms) in results. Once the terms filtered, we asked two domain experts to evaluate remain terms in order to determine the precision of our terminology extraction approach from log files. First extracted terms are tagged by a domain expert as *relevant* or *not relevant* according to the context and their usefulness in the logs. Then, another expert reviewed the tagged terms by the first expert.

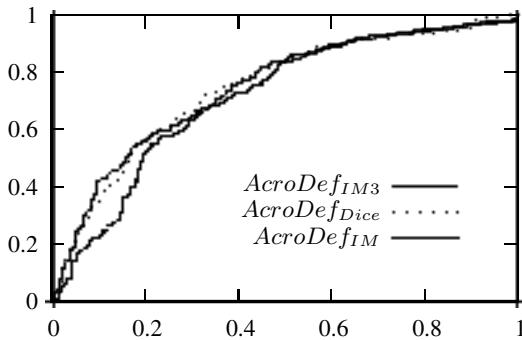
We evaluate the ranking function used to score the terms (*i.e.* *AcroDef*) using the ROC curves (Receiver Operating Curve) [26]. A ROC curve allows to compare the ranking functions (here *AcroDef*) that classify elements of a data set into the both groups, *i.e.* *positive* and *negative*. In our case, ROC curve indicates the ability of *AcroDef* to give a higher score to relevant terms than the irrelevant ones. Using ROC curves, we evaluate how much *AcroDef* is relevant as a measure to distinguish the positive and negative terms.

Figure 1 presents the ROC curves obtained based on three types of *AcroDef* with  $m = 700$ . To better analyse the ROC curves, we calculate the AUC (Area Under Curve) which is a synthetic indicator derived from the ROC curve. AUC is the area between the curve and the horizontal axis. If we order individuals at random, the AUC will be equal to 0.5.

We calculate AUC according to different thresholds of filtering. That is, the number of top-ranked terms by *AcroDef* which are selected as relevant. We consider 6 thresholds ( $m = 200, m = 300, m = 400, \dots, m = 700$ ).

The tables 1 shows AUC according to the ROC curves based on  $AcroDef_{MI}$ ,  $AcroDef_{MI3}$ , and  $AcroDef_{Dice}$ . According to AUC values, for example, when we use  $AcroDef_{MI3}$  and with  $m = 500$ , it is 74% probable that relevant terms have higher *AcroDef* score than irrelevant terms.

The results show that the best ranking function to evaluate and order the extracted terms is  $AcroDef_{MI3}$ .



**Fig. 1.** ROC curves based on three types of *AcroDef* and  $m = 500$

**Table 1.** AUC obtained at each level of filtering based on the three kind of *AcroDef*

$m$	$AUC_{MI}$	$AUC_{MI3}$	$AUC_{Dice}$
200	0.53	<b>0.60</b>	0.59
300	0.61	<b>0.70</b>	0.66
400	0.62	<b>0.71</b>	0.68
500	0.66	<b>0.74</b>	0.71
600	0.72	<b>0.75</b>	<b>0.75</b>
700	0.74	<b>0.77</b>	0.76

## 5 Conclusion and Future Work

In this paper, we describe a particular type of textual data: log files generated by tools for integrated circuit design. Since these log files are multi-source, multi-format, heterogeneous, and evolving textual data, the NLP and IE methods are not necessarily well suited to extract information.

To extract domain terminology, we extracted the co-occurrences. For that, we apply the specific preprocessing, normalization and tagging methods. To reduce the noise ratio in extracted terms and favor more relevant terms of this domain, we score terms using a Web and context based measure. Then, we select the most ranked terms by filtering based on score of terms. The experiments show that our approach of terminology extraction from log files, EXTERLOG, can achieve an F-score equal to 0.79 after filtering of terms.

To improve the performance of terminology extraction, we will develop our normalization method. Given the importance of accurate grammatical tagging, we will improve the grammatical tagger. Finally, we plan to take into account the terminology extracted using our system to enrich the patterns of information extraction from log files.

## References

1. Saneifar, H., Bonniol, S., Laurent, A., Poncelet, P., Roche, M.: Terminology extraction from log files. In: Bhowmick, S.S., Küng, J., Wagner, R. (eds.) DEXA 2009. LNCS, vol. 5690, pp. 769–776. Springer, Heidelberg (2009)
2. Yamanishi, K., Maruyama, Y.: Dynamic syslog mining for network failure monitoring. In: KDD 2005: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 499–508. ACM, New York (2005)
3. Facca, F.M., Lanzi, P.L.: Mining interesting knowledge from weblogs: a survey. *Data Knowl. Eng.* 53(3), 225–241 (2005)
4. Voorhees, E.M.: Query expansion using lexical-semantic relations. In: SIGIR 1994: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 61–69. Springer, Heidelberg (1994)
5. Dey, L., Singh, S., Rai, R., Gupta, S.: Ontology aided query expansion for retrieving relevant texts. In: Szczepaniak, P.S., Kacprzyk, J., Niewiadomski, A. (eds.) AWIC 2005. LNCS (LNAI), vol. 3528, pp. 126–132. Springer, Heidelberg (2005)
6. Even, F., Enguehard, C.: Extraction d'informations à partir de corpus dégradés. In: Proceedings of 9ème Conference sur le Traitement Automatique des Langues Naturelles (TALN 2002), pp. 105–115 (2002)
7. Mollá, D., Vicedo, J.L.: Question answering in restricted domains: An overview. *Computational Linguistics* 33(1), 41–61 (2007)
8. Roche, M., Heitz, T., Matte-Tailliez, O., Kodratoff, Y.: EXIT: Un système itératif pour l'extraction de la terminologie du domaine à partir de corpus spécialisés. In: Proceedings of JADT 2004 (International Conference on Statistical Analysis of Textual Data), vol. 2, pp. 946–956 (2004)
9. meng Tan, C., fang Wang, Y., do Lee, C.: The use of bigrams to enhance text categorization. In: Inf. Process. Manage., pp. 529–546 (2002)
10. Grobelnik, M.: Word sequences as features in text-learning. In: Proceedings of the 17th Electrotechnical and Computer Science Conference (ERK 1998), pp. 145–148 (1998)
11. David, S., Plante, P.: De la nécessité d'une approche morpho-syntaxique en analyse de textes. *Intelligence Artificielle et Sciences Cognitives au Québec* 2(3), 140–155 (1990)
12. Collier, N., Nobata, C., Tsujii, J.: Automatic acquisition and classification of terminology using a tagged corpus in the molecular biology domain. *Journal of Terminology*, John Benjamins 7(2), 239–257 (2002)
13. Lin, D.: Extracting collocations from text corpora. In: First Workshop on Computational Terminology, pp. 57–63 (1998)

14. Smadja, F.: Retrieving collocations from text: Xtract. Comput. Linguist. 19(1), 143–177 (1993)
15. Bourigault, D., Fabre, C.: Approche linguistique pour l’analyse syntaxique de corpus. Cahiers de Grammaire - Université Toulouse le Mirail (25), 131–151 (2000)
16. Daille, B.: Conceptual structuring through term variations. In: Proceedings of the ACL 2003, Workshop on Multiword Expressions, Morristown, NJ, USA, Association for Computational Linguistics, pp. 9–16 (2003)
17. Evans, D.A., Zhai, C.: Noun-phrase analysis in unrestricted text for information retrieval. In: Proceedings of the 34th Annual Meeting on Association for Computational Linguistics, Morristown, NJ, USA, Association for Computational Linguistics, pp. 17–24 (1996)
18. Brill, E.: A simple rule-based part of speech tagger. In: Proceedings of the Third Conference on Applied Natural Language Processing, pp. 152–155 (1992)
19. Amrani, A., Kodratoff, Y., Matte-Tailliez, O.: A semi-automatic system for tagging specialized corpora. In: Dai, H., Srikant, R., Zhang, C. (eds.) PAKDD 2004. LNCS (LNAI), vol. 3056, pp. 670–681. Springer, Heidelberg (2004)
20. Daille, B.: Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. In: The Balancing Act: Combining Symbolic and Statistical Approaches to Language, pp. 49–66. MIT Press, Cambridge (1996)
21. Church, K.W., Hanks, P.: Word association norms, mutual information, and lexicography. Computational Linguistics 16, 22–29 (1990)
22. Daille, B.: Approche mixte pour l’extraction automatique de terminologie: statistiques lexicales et filtres linguistiques. PhD thesis, Universit Paris 7 (1994)
23. Roche, M., Prince, V.: AcroDef: A quality measure for discriminating expansions of ambiguous acronyms. In: Kokinov, B., Richardson, D.C., Roth-Berghofer, T.R., Vieu, L. (eds.) CONTEXT 2007. LNCS (LNAI), vol. 4635, pp. 411–424. Springer, Heidelberg (2007)
24. Smadja, F., McKeown, K.R., Hatzivassiloglou, V.: Translating collocations for bilingual lexicons: A statistical approach. Computational Linguistics 22(1), 1–38 (1996)
25. Salton, G., Buckley, C.: Term weighting approaches in automatic text retrieval. Technical report, Ithaca, NY, USA (1987)
26. Flach, P., Blockeel, H., Ferri, C., Hernández-Orallo, J., Struyf, J.: Decision support for data mining: An introduction to ROC analysis and its applications. In: Data Mining and Decision Support: Integration and Collaboration pages, pp. 81–90. Kluwer, Dordrecht (2003)

# Average Cluster Consistency for Cluster Ensemble Selection

F. Jorge F. Duarte<sup>1</sup>, João M.M. Duarte<sup>1,2</sup>,  
Ana L.N. Fred<sup>2</sup>, and M. Fátima C. Rodrigues<sup>1</sup>

<sup>1</sup> GECAD - Knowledge Engineering and Decision Support Group

Instituto Superior de Engenharia do Porto

R. Dr. António Bernardino de Almeida, 431, P-4200-072 Porto, Portugal

{fjd, jod, fr}@isep.ipp.pt

www.gecad.isep.ipp.pt

<sup>2</sup> Instituto de Telecomunicações, Instituto Superior Técnico

Av. Rovisco Pais, 1, P-1049-001, Lisboa, Portugal

{jduarte, afred}@lx.it.pt

www.it.pt

**Abstract.** Various approaches to produce cluster ensembles and several consensus functions to combine data partitions have been proposed in order to obtain a more robust partition of the data. However, the existence of many approaches leads to another problem which consists in knowing which of these approaches to produce the cluster ensembles' data and to combine these partitions best fits a given data set. In this paper, we propose a new measure to select the best consensus data partition, among a variety of consensus partitions, based on the concept of average cluster consistency between each data partition that belongs to the cluster ensemble and a given consensus partition. The experimental results obtained by comparing this measure with other measures for cluster ensemble selection in 9 data sets, showed that the partitions selected by our measure generally were of superior quality in comparison with the consensus partitions selected by other measures.

## 1 Introduction

Data clustering goal consists of partitioning a data set into clusters, based on a concept of similarity between data so that similar data patterns are grouped together, and unlike patterns are separated into different clusters. Several clustering algorithms have been proposed in the literature but none can discover all kinds of cluster structures and shapes.

In order to improve data clustering robustness and quality [1], reuse clustering solutions [2] and cluster data in a distributed way, various cluster ensemble approaches have been proposed based on the idea of combining multiple data clustering results into a more robust and better quality consensus partition. The principal proposals to solve the cluster ensemble problem are based on: co-associations between pairs of patterns [3,4], mapping the cluster ensemble into graph [5], hyper-graph [2] or mixture model [6] formulations, and searching for a median partition that summarizes the cluster ensemble [7].

A cluster ensemble can be built by using different clustering algorithms [4], using distinct parameters and/or initializations to the same algorithm [3], sampling the original data set [8] and using different feature sets to produce each individual partition [9].

One can also apply different consensus functions to the same cluster ensemble. These variations in the cluster ensemble problem leads to a question: “*Which cluster ensemble construction method and which consensus function should one select for a given data set?*”. This paper addresses the implicit problem in the previous question by selecting the best consensus partition based on the concept of *average cluster consistency* between the consensus partition and the respective cluster ensemble.

The rest of this paper is organized as follows. In section 2, the cluster ensemble problem formulation (subsection 2.1), background work about cluster ensemble selection (subsection 2.2) and the clustering combination methods used in our experiments (subsection 2.3) are presented. Section 3 presents a new approach for cluster ensemble selection, based on the notion of average cluster consistency. The experimental setup used to assess the performance of our proposal is described in section 4 and the respective results are presented in section 5. Finally, the conclusions appear in section 6.

## 2 Background

### 2.1 Cluster Ensemble Formulation

Let  $\mathcal{X} = \{x_1, \dots, x_n\}$  be a set of  $n$  data patterns and let  $P = \{C_1, \dots, C_K\}$  be a partition of  $\mathcal{X}$  into  $K$  clusters. A cluster ensemble  $\mathcal{P}$  is defined as a set of  $N$  data partitions  $P^l$  of  $\mathcal{X}$ :

$$\mathcal{P} = \{P^1, \dots, P^N\}, P^l = \{C_1^l, \dots, C_{K^l}^l\}, \quad (1)$$

where  $C_k^l$  is the  $k^{\text{th}}$  cluster in data partition  $P^l$ , which contains  $K^l$  clusters, and  $\sum_{k=1}^{K^l} |C_k^l| = n$ ,  $\forall l \in \{1, \dots, N\}$ .

There are two fundamental phases in combining multiple data partitions: the partition generation mechanism and the consensus function, that is, the method that combines the  $N$  data partitions in  $\mathcal{P}$ . As introduced before, there are several ways to generate a cluster ensemble  $\mathcal{P}$ , such as, producing partitions of  $\mathcal{X}$  using different clustering algorithms, changing parameters and/or initializations for the same clustering algorithm, using different subsets of data features or patterns, projecting  $\mathcal{X}$  to subspaces and combinations of these. A consensus function  $f$  maps a cluster ensemble  $\mathcal{P}$  into a consensus partition  $P^*$ ,  $f : \mathcal{P} \rightarrow P^*$ , such that  $P^*$  should be robust and consistent with  $\mathcal{P}$ , i.e., the consensus partition should not change (significantly) when small variations are introduced in the cluster ensemble and the consensus partition should reveal the underlying structure of  $\mathcal{P}$ .

### 2.2 Cluster Ensemble Selection

As previously referred, the combination of multiple data partitions can be carried out in various ways, which may lead to very different consensus partitions. This diversity causes the problem of picking the best consensus data partition from all the produced ones.

In [10] work, a study was conducted on the diversity of the cluster ensemble and its relation to the consensus partition quality. Four measures were defined in order to assess the diversity of a cluster ensemble, by comparing each data partition  $P^l \in \mathcal{P}$  with the final data partition  $P^*$ . The adjusted Rand index [11] was used to assess the agreement between pairs of data clusterings ( $\text{Rand}(P^l, P^*) \in [0, 1]$ ). Values close to 1 mean that the clusterings are similar.

The first measure,  $\text{Div}_1(P^*, \mathcal{P})$ , is defined as the average diversity between each clustering  $P^l \in \mathcal{P}$  and the consensus partition  $P^*$ . The diversity between  $P^l$  and  $P^*$  is defined as  $1 - \text{Rand}(P^l, P^*)$ . Formally, the average diversity between  $P^*$  and  $\mathcal{P}$  is defined as:

$$\text{Div}_1(P^*, \mathcal{P}) = \frac{1}{N} \sum_{l=1}^N 1 - \text{Rand}(P^l, P^*). \quad (2)$$

Previous work [12] showed that the cluster ensembles that exhibit higher individual variation of diversity generally obtained better consensus partitions.

The second measure,  $\text{Div}_2(P^*, \mathcal{P})$ , is based on this idea and is defined as the standard deviation of cluster ensemble individual diversity:

$$\text{Div}_2(P^*, \mathcal{P}) = \sqrt{\frac{1}{N-1} \sum_{l=1}^N (1 - \text{Rand}(P^l, P^*) - \text{Div}_1)^2}, \quad (3)$$

where  $\text{Div}_1$  is  $\text{Div}_1(P^*, \mathcal{P})$ .

The third diversity measure,  $\text{Div}_3(P^*, \mathcal{P})$  is based on the intuition that the consensus partition,  $P^*$ , is similar to the *real* structure of the data set. So, if the clusterings  $P^l \in \mathcal{P}$  are similar to  $P^*$ , i.e.,  $1 - \text{Div}_1$  is close to 1,  $P^*$  is expected to be a high quality consensus partition. Nevertheless, as it is assumed that cluster ensembles with high individual diversity variance are likely to produce good consensus partitions, the third measure also includes a component associated to  $\text{Div}_2(P^*, \mathcal{P})$ . It is formally defined as:

$$\text{Div}_3(P^*, \mathcal{P}) = \frac{1}{2}(1 - \text{Div}_1 + \text{Div}_2), \quad (4)$$

where  $\text{Div}_2$  corresponds to  $\text{Div}_2(P^*, \mathcal{P})$ .

The forth measure,  $\text{Div}_4(P^*, \mathcal{P})$ , simply consists of a ratio between the standard deviation of the cluster ensemble individual diversity and the average diversity between  $P^*$  and  $\mathcal{P}$ , as shown in equation 5.

$$\text{Div}_4(P^*, \mathcal{P}) = \frac{\text{Div}_2(P^*, \mathcal{P})}{\text{Div}_1(P^*, \mathcal{P})} \quad (5)$$

The four previously referred measures were compared in [10] and the authors concluded that only  $\text{Div}_1(P^*, \mathcal{P})$  and, specially,  $\text{Div}_3(P^*, \mathcal{P})$  measures showed some correlation with the quality of the consensus partition. Despite that, in some data sets the quality of the final data partitions increased as  $\text{Div}_1(P^*, \mathcal{P})$  and  $\text{Div}_3(P^*, \mathcal{P})$  also increased, in several other data sets it did not occur. The authors recommended that one should select the cluster ensembles with the median values of  $\text{Div}_1(P^*, \mathcal{P})$  or  $\text{Div}_3(P^*, \mathcal{P})$  to choose a good consensus partition.

In other work [2], the best consensus partition  $P^B$  is thought as the consensus partition  $P^*$  that maximizes the Normalized Mutual Information (NMI) between each data partition  $P^l \in \mathcal{P}$  and  $P^*$ , i.e.,  $P^B = \arg \max_{P^*} \sum_l^N NMI(P^*, P^l)$ .  $NMI(P^*, P^l)$  is defined as:

$$NMI(P^*, P^l) = \frac{MI(P^*, P^l)}{\sqrt{H(P^*)H(P^l)}}, \quad (6)$$

where  $MI(P^*, P^l)$  is the mutual information between  $P^*$  and  $P^l$  (eq. 7) and  $H(P)$  is the entropy of  $P$  (eq. 8). The mutual information between two data partitions,  $P^*$  and  $P^l$ , is defined as:

$$MI(P^*, P^l) = \sum_i^{K^*} \sum_j^{K^l} \frac{Prob(i, j)}{Prob(i)Prob(j)}, \quad (7)$$

with  $Prob(k) = \frac{n_k}{n}$ , where  $n_k$  is the number of patterns in the  $k^{\text{th}}$  cluster of  $P$ , and  $Prob(i, j) = \frac{1}{n}|C_i^* \cap C_j^l|$ .

The entropy of a data partition  $P$  is given by:

$$H(P) = - \sum_{k=1}^K Prob(k) \log Prob(k). \quad (8)$$

Therefore, the Average Normalized Mutual Information ( $ANMI(P^*, \mathcal{P})$ ) between the cluster ensemble and a consensus partition, defined in eq. 9, can be used to select the best consensus partition. Higher values of  $ANMI(P^*, \mathcal{P})$  suggest better quality consensus partitions.

$$ANMI(P^*, \mathcal{P}) = \frac{1}{N} \sum_{l=1}^N NMI(P^*, P^l). \quad (9)$$

### 2.3 WEACS

The Weighted Evidence Accumulation Clustering using Subsampling (WEACS) [4] approach is an extension to Evidence Accumulation Clustering (EAC) [1]. EAC considers each data partition  $P^l \in \mathcal{P}$  as an independent evidence of data organization. The underlying assumption of EAC is that two patterns belonging to the same *natural* cluster will be frequently grouped together. A vote is given to a pair of patterns every time they co-occur in the same cluster. Pairwise votes are stored in a  $n \times n$  co-association matrix and are normalized by the total number of combining data partitions:

$$co\_assoc_{ij} = \frac{\sum_{l=1}^N vote_{ij}^l}{N}, \quad (10)$$

where  $vote_{ij}^l = 1$  if  $x_i$  and  $x_j$  belong to the same cluster  $C_k^l$  in the data partition  $P^l$ , otherwise  $vote_{ij}^l = 0$ . In order to produce the consensus partition, one can apply any clustering algorithm over the co-association matrix  $co\_assoc$ .

WEACS extends EAC by weighting each pattern pairwise vote based on the quality of each data partition  $P^l$  and by using subsampling in the construction of the cluster ensemble. The idea consists of perturbing the data set and assigning higher relevance to better data partitions in order to produce better combination results. To weight each  $vote_{ij}^l$  in a weighted co-association matrix,  $w\_co\_assoc$ , one or several internal clustering validity indices are used to measure the quality of each data partition  $P^l$ , and the corresponding normalized index value,  $IV^l$ , corresponds to the weight factor. Note that the internal validity indices assess the clustering results in terms of quantities that involve only the features of the data set, so no *a priori* information is provided. Formally,  $w\_co\_assoc$  is defined as

$$w_{co\_assoc_{ij}} = \frac{\sum_{l=1}^N IV^l \times vote_{ij}^l}{S_{ij}}, \quad (11)$$

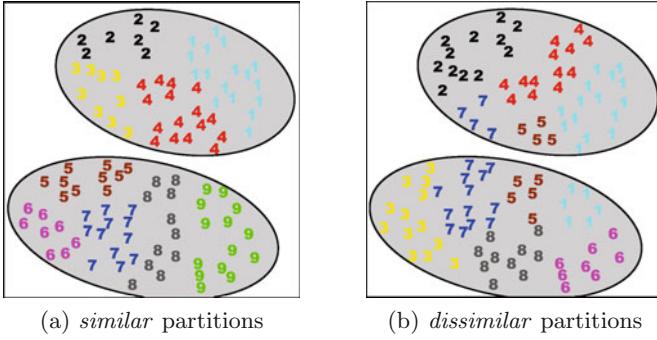
where  $S$  is a  $n \times n$  matrix with  $S_{ij}$  equal to the number of data partitions where both  $x_i$  and  $x_j$  are simultaneously selected to belong to the same data subsample.

There are two versions of WEACS that correspond to two different ways for computing the weight factor  $IV^l$ . The first one, Single WEACS (SWEACS), uses the result of only one clustering validity index to assess the quality of  $P^l$ , i.e.,  $IV^l = norm\_validity(P^l)$ , where  $norm\_validity(\cdot)$  corresponds to a normalized validity index function that returns a value in the interval  $[0, 1]$ . Higher values correspond to better data partitions. In the second version, Joint WEACS (JWEACS),  $IV^l$  is defined as the average of the output values of  $NumInd$  normalized validity index functions,  $norm\_validity_m(\cdot)$ , applied to  $P^l$ , i.e.,  $IV^l = \sum_{m=1}^{NumInd} norm\_validity_m(P^l)/NumInd$ .

In the WEACS approach, one can use different cluster ensemble construction methods, different clustering algorithms to obtain the consensus partition, and, particularly in the SWEACS version, one can even use different cluster validity indices to weight pattern pairwise votes. These constitute variations of the approach, taking each of the possible modifications as a configuration parameter of the method. As shown in section 4, although the WEACS leads in general to good results, no individual tested configuration led consistently to the best result in all data sets. We used a complementary step to the WEACS approach which consists of combining all the final data partitions obtained in the WEACS approach within a cluster ensemble construction method using EAC. The interested reader is encouraged to read [4] for a detailed description of WEACS.

### 3 Average Cluster Consistency (ACC)

The idea behind Average Cluster Consistency (ACC) measure [13] is that if the similarity between the multiple data partitions in the cluster ensemble and the consensus partition is high, the quality of the consensus partition will also be high. Some clustering combination methods, such as the EAC and WEACS methods presented in subsection 2.3, usually produce better quality consensus data partitions when combining data partitions with more clusters than the expected *real* number of clusters  $K^0$ . This difference in the number of clusters usually leads to low similarity scores when comparing two data partitions. For this reason, a new concept for comparing data partitions was defined. In this new similarity measure between two data partitions,  $P^l$  and  $P^0$  with

**Fig. 1.** Example of Average Cluster Consistency motivation

$K^l >> K^0$ , if each of the  $K^l$  clusters  $C_k^l \in P^l$  is a subset of a cluster  $C_m^0 \in P^0$ , i.e.  $C_k^l \subseteq C_m^0$ , then the partitions  $P^l$  and  $P^0$  have the maximum degree of similarity. If the data patterns belonging to each cluster in  $P^l$  are split into different clusters in  $P^0$ , the data partitions  $P^l$  and  $P^0$  are dissimilar. Figure 1 shows an example of the previously described situations. The figure includes two consensus partitions (one in figure 1 (a) and another in figure 1 (b)) each with  $K^0 = 2$  clusters (shaded areas). Inside each consensus partition's clusters, there are several patterns represented by numbers, which indicate the cluster labels assigned to the data patterns in a partition  $P^l$  belonging to the cluster ensemble. Note that the number of clusters of the partition  $P^l$  is higher than the number of clusters of the consensus partition  $P^0$  ( $K^l >> K^0$ ). On the left figure, a perfect similarity between  $P^0$  and  $P^l$  is presented as all data patterns of each cluster  $C_k^l$  belong to the same cluster in  $P^0$ . On the right figure, two dissimilar partitions are presented as the data patterns belonging to clusters 1, 5 and 7 in  $P^l$  are divided in the two clusters of  $P^0$ .

Our similarity measure between two partitions,  $P^*$  and  $P^l$ , is then defined as

$$\text{sim}(P^*, P^l) = \frac{\sum_{m=1}^{K^l} \max_{1 \leq k \leq K^*} |\text{Inters}_{km}| (1 - \frac{|C_k^*|}{n})}{n}, \quad (12)$$

where  $K^l \geq K^*$ ,  $|\text{Inters}_{km}|$  is the cardinality of the set of patterns common to the  $k^{\text{th}}$  and  $m^{\text{th}}$  clusters of  $P^*$  and  $P^l$ , respectively ( $\text{Inters}_{km} = \{x_a | x_a \in C_k^* \wedge x_a \in C_m^l\}$ ). Note that in Eq. 12,  $|\text{Inters}_{km}|$  is weighted by  $(1 - \frac{|C_k^*|}{n})$  in order to prevent cases where  $P^*$  has clusters with almost all data patterns and would obtain a high value of similarity.

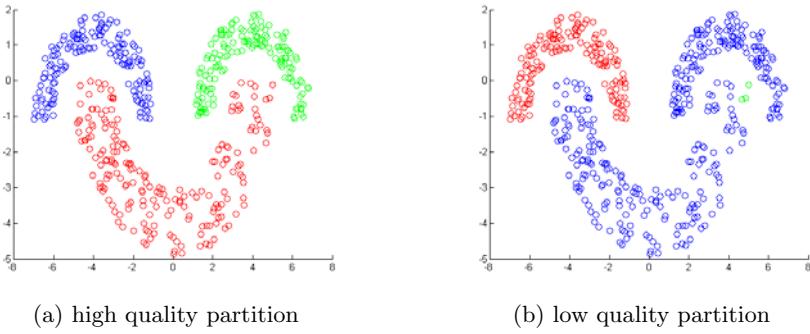
The Average Cluster Consistency measures the average similarity between each data partition in the cluster ensemble ( $P^l \in \mathcal{P}$ ) and a target consensus partition  $P^*$ , using the previously explained notion of similarity. It is formally defined by

$$ACC(P^*, \mathcal{P}) = \frac{\sum_{i=1}^N \text{sim}(P^i, P^*)}{N}. \quad (13)$$

From a set of possible choices, the *best* consensus partition is the one that achieves the highest  $ACC(P^*, \mathcal{P})$  value. Note that by the fact of using subsampling, the ACC

**Table 1.** ACC values obtained for the data partitions shown in figure 2 with and without the use of the weighting factor

Data set partitions	Partition a	Partition b
ACC not using the weighting factor	1.0000	1.0000
ACC using the weighting factor	0.6595	0.4374



**Fig. 2.** Two data partitions of the Half Rings data set

measure only uses the data patterns of the consensus partition  $P^*$  that appear in the combining data partition  $P^l \in \mathcal{P}$ .

In order to justify the use of the weighting factor  $\left(1 - \frac{|C_k^*|}{n}\right)$  in our similarity measure between two data partitions (equation 12) used in the ACC measure (equation 13), we present the example shown in figure 2. This figure shows two consensus partitions of a synthetic data set used in our experiments, the Half Rings data set (presented in section 4). Both consensus partitions have 3 clusters and were obtained using two different clustering algorithms (Single-Link and K-means) to extract the consensus partition in the WEACS approach.

The first consensus partition is perfect since it correctly identifies the three existing groups in the data set, while the second consensus partition is of poor quality because it contains a large cluster (represented in blue) that almost encompasses two real clusters of the data set.

Table 1 shows the values obtained by ACC measure without (second line) and with (third line) the use of the weighting factor for both data partitions. The ACC measure without the use of the weighting factor obtained the value 1 for both data partitions, while the ACC measure using the weighting factor obtained the value 0.6595 for the “optimal” partition (figure 2 a) and 0.4374 for the other partition (figure 2 b). As can be seen by this example, the use of the weighting factor in our similarity measure between two data partition (equation 12) prevents cases where the consensus partitions have clusters with almost all data patterns and would obtain a high value of similarity.

At the first glance, this measure may seem to contradict the observations by [10] and [12] which point out that the clustering quality is improved with the increase of diversity in the cluster ensemble. However, imagine that each data partition belonging to a cluster ensemble is obtained by random guess. The resulting cluster ensemble is very

diverse but does not provide useful information about the structure of the data set, so, it is expected to produce a low quality consensus partition. For this reason, one should distinguish the “good” diversity from the “bad” diversity. Our definition of similarity between data partitions (Eq. 12) considers that two apparently different data partitions (for instance, partitions with different number of clusters) may be similar if they have a common structure, as shown in the figure 1 (a) example, and the outcome is the selection of cluster ensembles with “good” diversity rather than the ones with “bad” diversity.

## 4 Experimental Setup

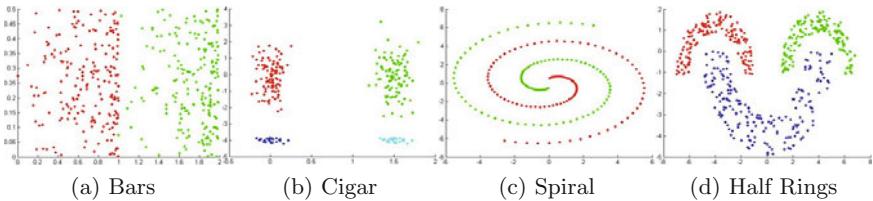
We used 4 synthetic and 5 real data sets to assess the quality of the cluster ensemble methods on a wide variety of situations, such as data sets with different cardinality and dimensionality, arbitrary shaped clusters, well separated and touching clusters and distinct cluster densities. A brief description for each data set is given below.

**Synthetic Data Sets.** Fig. 3 presents the 2-dimensional synthetic data sets used in our experiments. Bars data set is composed by two clusters very close together, each with 200 patterns, with increasingly density from left to right. Cigar data set consists of four clusters, two of them having 100 patterns each and the other two groups 25 patterns each. Spiral data set contains two spiral shaped clusters with 100 data patterns each. Half Rings data set is composed by three clusters, two of them have 150 patterns and the third one 200.

**Real Data Sets.** The 5 real data sets used in our experiments are available at UCI repository (<http://mlearn.ics.uci.edu/MLRepository.html>). The first one is Iris and consists of 50 patterns from each of three species of Iris flowers (setosa, virginica and versicolor) characterized by four features. One of the clusters is well separated from the other two overlapping clusters. Breast Cancer data set is composed of 683 data patterns characterized by nine features and divided into two clusters: benign and malignant. Yeast Cell data set consists of 384 patterns described by 17 attributes, split into five clusters concerning five phases of the cell cycle. There are two versions of this dataset, the first one is called Log Yeast and uses the logarithm of the expression level and the other is called Std Yeast and is a “standardized” version of the same data set, with mean 0 and variance 1. Finally, Optdigits is a subset of Handwritten Digits data set containing only the first 100 objects of each digit, from a total of 3823 data patterns characterized by 64 attributes.

In order to produce the cluster ensembles, we applied the Single-Link (SL) [14], Average-Link (AL) [14], Complete-Link (CL) [15], K-means (KM) [16], CLARANS (CLR) [17], Chameleon (CHM) [18], CLIQUE [19], CURE [20], DBSCAN [21] and STING [22] clustering algorithms to each data set to generate 50 cluster ensembles for each clustering algorithm. Each cluster ensemble has 100 data partitions with the number of clusters,  $K$ , randomly chosen in the set  $K \in \{10, \dots, 30\}$ .

After all cluster ensembles have been produced, we applied the EAC, SWEACS and JWEACS approaches using the KM, SL, AL and Ward-Link (WR) [23] clustering algorithms to produce the consensus partitions. The number of clusters of the combined data partitions were set to be the *real* number of clusters of each data set. We also defined



**Fig. 3.** Synthetic data sets

other two cluster ensembles: ALL5 and ALL10. The cluster ensemble referred as ALL5 is composed by the data partitions of SL, AL, CL, KM and CLR algorithms ( $N = 500$ ) and the cluster ensemble ALL10 is composed by the data partitions produced by all data clustering algorithms ( $N = 1000$ ).

To evaluate the quality of the consensus partitions we used the Consistency index ( $C_i$ ) [1].  $C_i$  measures the fraction of shared data patterns in matching clusters of the consensus partition ( $P^*$ ) and of the *real* data partition ( $P^0$ ). Formally, the Consistency index is defined as

$$C_i(P^*, P^0) = \frac{1}{n} \sum_{k=1}^{\min\{K^*, K^0\}} |C_k^* \cap C_k^0| \quad (14)$$

where  $|C_k^* \cap C_k^0|$  is the cardinality of the  $P^*$  and  $P^0$   $k^{\text{th}}$  matching clusters data patterns intersection.

As an example, table 2 shows the results of the cluster combination approaches for the Optdigits data set, averaged over the 50 runs. In this table, rows are grouped by cluster ensemble construction method. Inside each cluster ensemble construction method appears the 4 clustering algorithms used to extract the final data partition (KM, SL, CL and WR). The last column (C. Step) shows the results of the complementary step of WEACS. As it can be seen, the results vary from a very poor result obtained by SWEACS, combining data partitions produced by SL algorithm and using the K-means algorithm to extract the consensus partitions (10% of accuracy), to good results obtained by all clustering combination approaches, when combining data partitions produced by CHM and using the WR algorithm to extract the consensus partition. For this configuration, EAC achieved 87.54% of accuracy, JWEAC 87.74%, SWEAC 87.91% using PS validity index to weight each vote in  $w\_co\_assoc$ , and 88.03% using the complementary step. Due to space restrictions and to the fact that this is not the main topic of this paper, we do not present the results for the other data sets used in our experiments.

Table 3 shows the average and best  $C_i(P^*, P^0)$  percentage values obtained by each clustering combination method for each data set. We present this table to remark that the average quality of the consensus partitions produced by each clustering combination method is substantially different from the best ones. As an example, SWEACS approach achieved 90.89% as the best result for Std Yeast data set while the average accuracy was only of 54.00%.

The results presented in tables 2 and 3 show that different cluster ensemble construction methods and consensus functions can produce consensus partitions with very

**Table 2.** Average  $C_i(P^*, P^0)$  percentage values obtained by EAC, JWEACS and SWEACS for Optdigits data set

CE	Ext.Alg	EAC	JWEAC	Hub	Dunn	S.,Dbn	CH	S	I	XB	DB	SD	PS	C.Step
SL	KM	39.75	34.47	36.89	36.66	38.14	35.29	10.00	39.16	38.03	33.84	42.09	33.55	34.19
	SL	10.60	10.60	10.60	10.60	10.60	10.10	10.10	10.60	10.60	10.60	10.60	10.60	11.19
	AL	10.60	10.60	10.60	10.60	10.60	10.60	10.10	10.60	10.60	10.60	10.60	10.60	20.21
	WR	40.31	40.31	40.53	40.30	40.40	40.31	10.10	40.30	40.31	40.40	40.49	40.31	44.28
AL	KM	70.33	69.84	71.09	68.83	70.40	71.47	70.42	72.19	69.59	67.68	69.49	68.83	73.93
	SL	60.14	60.14	60.14	51.48	60.37	60.14	60.37	60.14	60.14	60.14	60.14	60.14	67.65
	AL	67.29	67.28	67.29	67.29	67.30	67.29	69.42	67.28	67.29	67.29	67.29	67.28	82.10
	WR	82.06	82.10	82.10	83.57	84.31	82.10	84.31	82.10	82.10	82.10	82.09	84.32	84.32
CL	KM	62.77	62.39	64.20	63.05	62.28	64.97	64.82	66.30	62.97	63.78	68.95	62.92	64.25
	SL	53.76	52.54	53.80	53.80	53.80	58.45	58.57	58.25	52.72	53.80	52.47	52.52	58.15
	AL	69.28	70.97	70.94	70.94	69.28	70.89	71.21	63.50	69.28	70.94	70.94	70.94	70.53
	WR	76.27	76.34	76.27	76.27	71.14	76.35	71.14	76.34	76.26	76.35	76.35	71.25	71.25
KM	KM	68.77	69.43	72.56	69.97	73.75	73.43	69.52	70.9	69.57	69.29	71.81	74.39	67.86
	SL	30.59	30.60	30.21	30.60	30.78	30.21	30.78	30.61	30.78	30.60	30.61	30.60	59.50
	AL	79.78	79.43	79.24	79.51	79.32	77.49	79.41	77.54	79.41	79.78	79.41	79.60	79.35
	WR	79.51	79.67	79.49	79.85	79.71	77.11	78.85	77.00	78.74	78.97	78.87	79.75	78.05
CLR	KM	63.96	63.61	65.60	65.24	65.39	67.14	64.58	65.13	62.32	65.69	62.28	65.38	62.81
	SL	20.31	20.11	20.31	20.51	20.51	19.81	20.31	19.81	20.40	20.31	20.31	20.31	42.67
	AL	82.73	82.37	82.24	82.78	82.48	75.53	81.11	75.32	82.60	82.21	82.85	79.34	76.15
	WR	78.85	78.66	79.27	79.25	77.54	78.58	79.37	78.81	79.06	78.86	77.12	79.27	77.37
ALL5	KM	71.49	69.85	69.52	69.93	69.43	71.31	69.67	70.70	75.98	70.57	69.11	67.77	64.77
	SL	39.50	30.30	49.24	30.30	20.81	40.40	49.83	40.39	30.39	20.60	30.30	30.30	51.23
	AL	65.57	65.22	73.21	51.24	30.50	71.14	80.44	65.62	60.11	30.41	30.60	30.79	65.32
	WR	80.86	80.51	80.89	80.89	80.76	80.95	80.54	80.99	80.53	80.31	80.69	80.51	80.85
CHM	KM	71.97	72.12	73.11	71.40	73.74	72.17	72.69	72.77	73.20	70.48	72.22	73.10	68.74
	SL	62.44	62.24	62.06	62.43	62.62	62.63	62.63	61.61	62.61	62.44	62.24	62.24	78.34
	AL	87.14	86.88	86.53	87.28	86.46	87.28	87.31	86.78	86.26	86.75	86.82	86.50	84.78
	WR	87.54	87.74	87.61	87.51	87.53	87.78	87.52	87.72	87.85	87.68	87.76	87.91	88.03
CLIQUE	KM	59.41	60.29	61.33	59.84	59.95	60.69	63.27	61.28	61.90	60.50	60.41	60.30	64.19
	SL	10.50	10.47	10.50	10.48	10.48	10.50	10.47	10.49	10.50	10.48	10.48	10.50	18.76
	AL	61.03	63.30	64.89	62.20	62.13	63.67	65.71	64.12	66.02	63.65	63.29	64.54	62.85
	WR	67.00	68.23	69.11	67.65	67.68	68.77	73.19	71.07	71.36	69.30	68.67	69.03	70.69
CURE	KM	58.84	57.03	62.75	58.15	45.17	66.12	23.81	51.28	50.60	55.22	52.17	46.88	63.06
	SL	10.63	10.63	10.63	10.62	10.62	10.61	10.61	10.63	10.63	10.63	10.63	10.63	11.00
	AL	10.60	10.60	10.58	10.60	10.61	10.63	18.39	10.61	10.60	10.61	10.61	10.60	26.81
	WR	67.09	67.04	75.55	68.00	62.29	77.48	26.16	71.46	63.41	65.81	63.82	63.56	71.25
DBSCAN	KM	68.81	69.61	70.18	67.85	66.97	69.71	68.68	68.51	69.42	69.04	69.51	70.00	71.10
	SL	62.87	62.56	63.01	63.15	62.72	64.40	62.52	65.09	63.88	63.16	62.84	63.20	75.86
	AL	77.21	77.16	77.07	77.11	76.76	76.90	77.16	77.25	76.69	77.20	76.85	76.88	77.32
	WR	80.98	79.84	80.02	80.36	81.06	79.13	80.78	78.82	78.83	80.61	79.99	79.36	81.19
STING	KM	60.60	59.77	59.00	59.49	60.27	60.09	58.60	59.01	58.70	59.17	59.47	58.53	62.07
	SL	22.03	22.17	22.05	21.99	22.59	19.59	23.71	22.50	22.01	22.01	22.02	22.02	34.97
	AL	37.89	38.01	37.86	38.07	36.32	39.97	46.09	42.06	37.97	36.72	37.67	37.60	48.40
	WR	57.65	57.74	59.70	57.60	57.66	57.69	66.12	57.77	57.72	57.64	57.70	57.63	58.35
ALL10	KM	72.36	72.05	72.50	72.64	72.04	71.40	72.33	72.36	72.62	73.39	72.99	73.67	66.39
	SL	42.66	38.14	53.27	53.91	20.63	55.39	55.24	49.65	30.82	20.47	30.20	30.21	59.59
	AL	74.22	70.63	74.95	61.66	22.04	76.03	83.09	75.23	62.20	30.59	30.23	31.40	73.58
	WR	83.24	83.87	83.65	83.80	83.83	83.14	83.78	82.89	84.14	83.54	84.19	83.69	83.16

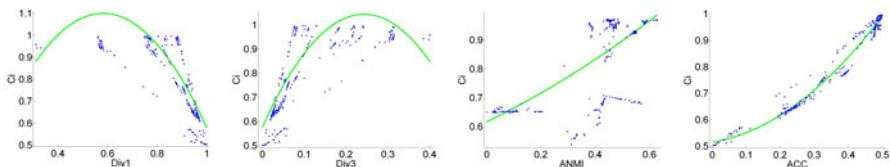
**Table 3.** Average and best  $C_i(P^*, P^0)$  percentage values obtained by EAC, JWEACS and SWEACS for all data sets

Approach	Bars	Breast	Cigar	Half Rings	Iris	Log Yeast	Optical	Std Yeast	Spiral
EAC	Average	86.80	80.96	85.57	84.13	73.88	34.14	58.33	53.23
	Best	99.50	97.07	100.00	100.00	97.37	40.93	87.54	88.50
SWEACS	Average	84.65	80.58	84.23	83.10	74.30	33.97	57.25	54.00
	Best	99.50	97.08	100.00	100.00	97.19	41.57	87.74	90.89
JWEACS	Average	86.98	80.88	84.66	83.96	74.59	34.16	57.83	53.80
	Best	99.50	97.20	100.00	100.00	97.29	41.58	87.91	92.64

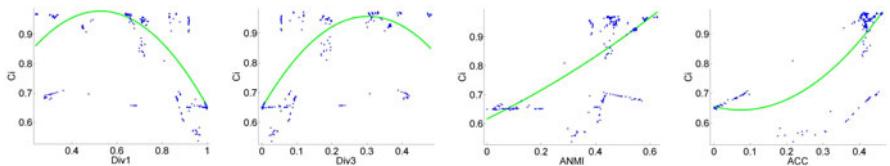
different quality. This reason emphasizes the importance of selecting the best consensus partition from a variety of possible consensus data partitions.

## 5 Results

In order to assess the quality of Average Cluster Consistency (ACC) measure (Eq. 13), we compared its performance against three other measures: the Average Normalized Mutual Information (ANMI) measure (Eq. 9), the  $Div_1$  measure (Eq. 2) and the  $Div_3$



**Fig. 4.**  $C_i$  vs each cluster ensemble selection measures for Bars data set



**Fig. 5.**  $C_i$  vs each cluster ensemble selection measures for Breast Cancer data set

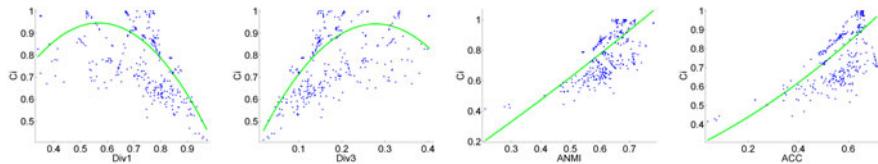
measure (Eq. 4). For each data set, the four measures were calculated for each consensus clustering produced by the clustering combination methods. These values were plotted (figures 4-12) against the respective clustering quality values of each consensus partition ( $C_i(P^*, P^0)$ ). Dots represent the consensus partitions, their positions in the horizontal axis represent the obtained values for the cluster ensemble selection measures and the corresponding positions in the vertical axis indicate the  $C_i$  values. The lines shown in the plots were obtained by polynomial interpolation of degree 2.

Figure 4 presents the results obtained by the cluster ensemble selection measures for Bars data set.  $Div_1$  values decrease with the increment of the quality of the consensus partitions, while the values of  $Div_3$  increase as the quality of the consensus partitions is improved. However, the correlations between  $Div_1$  with  $C_i$  and  $Div_3$  with  $C_i$  are not clearly evident. In the ANMI and ACC plots, one can easily see that as the values of these measures increase, the quality of the consensus partitions are improved.

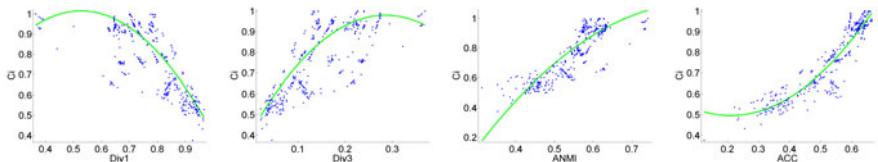
The results achieved for Breast Cancer data set are shown in figure 5. It can be seen that  $Div_1$  and  $Div_3$  measures are not correlated with the quality ( $C_i$  values) of the consensus partitions. However, in ANMI and ACC cluster ensemble selection measures there is a tendency of quality improvement as the values of these measures augment.

In the results obtained for Cigar data set, all the four measures showed some correlation with the Consistency index values (figure 6). For  $Div_1$  measure, the quality of the consensus partitions are improved as  $Div_1$  values decrease. For the remaining measures, the increasing of their values are followed by the improvement of the consensus partitions. Note that the dispersion of the dots in  $Div_1$  and  $Div_3$  plots are clearly higher than the dispersion presented in ANMI and ACC plots, showing that the correlations with  $C_i$  of the latter two measures are much stronger.

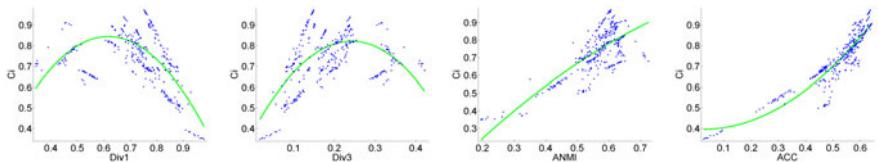
Figures 7 and 8 present the plots obtained for the selection of the best consensus partition for Half Rings and Iris data sets. The behavior of the measures are similar in both data sets and they are all correlated with the quality of the consensus partition. Again, one can see that as the values of  $Div_3$ , ANMI and ACC measures increase, the



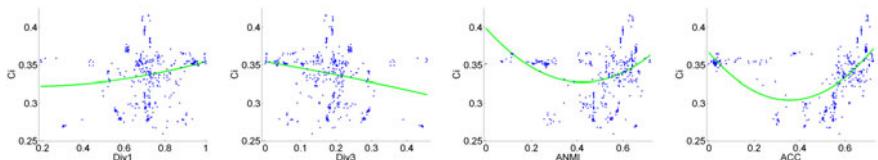
**Fig. 6.**  $C_i$  vs each cluster ensemble selection measures for Cigar data set



**Fig. 7.**  $C_i$  vs each cluster ensemble selection measures for Half Rings data set



**Fig. 8.**  $C_i$  vs each cluster ensemble selection measures for Iris data set

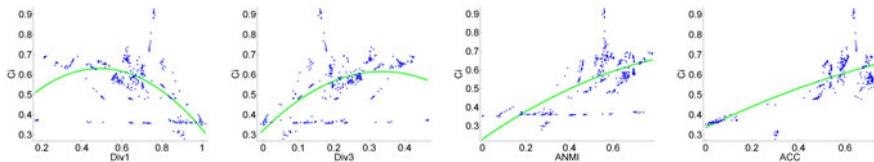


**Fig. 9.**  $C_i$  vs each cluster ensemble selection measures for Log Yeast data set

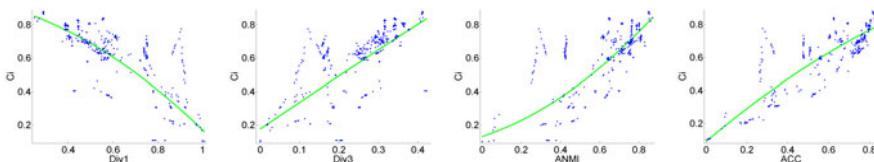
quality of the consensus partition is improved, while there is an inverse tendency for  $Div_1$  measure. In both data sets, the ACC measure is the one that better correlates its values with  $C_i$  as it is the one with the lowest dispersion of the dots in the plot.

The results for the Log Yeast data set are presented in figure 9. The  $Div_1$  and  $Div_3$  measures show no correlations with the quality of the consensus partitions. The ANMI and ACC measures also do not show a clear correlation with  $C_i$ . However, in both plots, one can see a cloud of dots that indicates some correlation between the measures and the Consistency index, specially in the ACC plot.

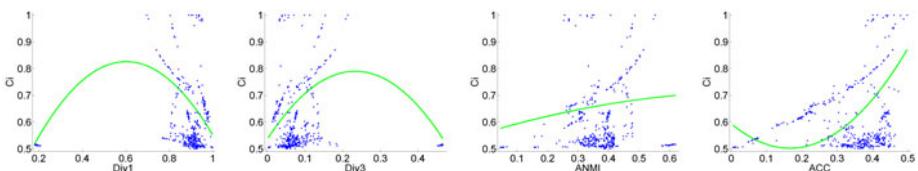
In figure 10, the results of the cluster ensemble selection methods for Std Yeast data set are presented. Once again, there is no clear correlation between  $Div_1$  and  $Div_3$  measures and the  $C_i$  values. The ANMI and ACC measures also do not present such correlation. However, there is a weak tendency of clustering quality improvement as these measures values increase.



**Fig. 10.**  $C_i$  vs each cluster ensemble selection measures for Std Yeast data set



**Fig. 11.**  $C_i$  vs each cluster ensemble selection measures for Optdigits data set



**Fig. 12.**  $C_i$  vs each cluster ensemble selection measure for Spiral data set

In the Optdigits data set, all measures are correlated with the quality of the consensus partitions. This correlation is stronger in ACC measure, as it can be seen in figure 11. The values of  $Div_1$  decrease as the clustering quality is improved while the quality of the consensus partitions is improved as the values of  $Div_3$ , ANMI and ACC measures increase.

The plots for the last data set, Spiral, are presented in figure 12. The  $Div_1$  and  $Div_3$  measures do not present correlation with  $C_i$  values, while the ANMI and ACC measures show weak tendencies of clustering improvement with the increasing of their values, specially in ACC cluster ensemble selection measure.

Table 4 shows the correlation coefficients between the Consistency index and the consensus partition selection measures. Values close to 1 (-1) suggest that there is a positive (negative) linear relationship between  $C_i$  and the selection measure, while values close to 0 indicate that there is no such linear relationship. In 6 out of the 9 data sets used in the experiments, the ACC measure obtained the highest linear relationship with the clustering quality (measured using the Consistency index). In the other 3 data sets, the highest linear relationships were obtained by the ANMI measure in the Bars (0.8635 against 0.8480 achieved by ACC) and Cigar (0.6293 against 0.6154 achieved by ACC) data sets, and by the  $Div_3$  measure in the Log Yeast data set which achieved -0.2820, a counterintuitive correlation coefficient when observing the positive coefficients obtained by  $Div_3$  for all the other data sets. In average, the ACC measure presents the

**Table 4.** Correlation coefficients between the Consistency index ( $C_i$ ) and the consensus partition selection measures ( $Div_1$ ,  $Div_3$ , ANMI and ACC measures) for each data set

Measure	Bars	Breast C.	Cigar	Half Rings	Iris	Log Yeast	Std Yeast	Optdigits	Spiral	Average
$Div_1$	-0.5712	-0.6006	-0.3855	-0.6444	-0.3010	0.2448	-0.5356	-0.7922	0.0044	-0.3979
$Div_3$	0.6266	0.6487	0.4367	0.6838	0.2578	<b>-0.2820</b>	0.5450	0.7123	0.0450	0.4082
ANMI	<b>0.8635</b>	0.7979	<b>0.6293</b>	0.8480	0.6856	-0.0444	0.7141	0.7785	0.1095	0.5980
ACC	0.8480	<b>0.8684</b>	0.6154	<b>0.9308</b>	<b>0.8785</b>	-0.0897	<b>0.8505</b>	<b>0.9149</b>	0.4187	<b>0.6928</b>

**Table 5.**  $C_i$  values for the consensus partition selected by  $Div_1$ ,  $Div_3$ , ANMI and ACC measures, and the maximum  $C_i$  value obtained, for each data set

Measure	Bars	Breast C.	Cigar	Half Rings	Iris	Log Yeast	Std Yeast	Optdigits	Spiral	Average
$Div_1$	95.47	95.11	97.93	99.90	87.35	26.96	57.97	58.55	51.68	74.54
$Div_3$	<b>99.50</b>	95.38	<b>100.0</b>	<b>100.0</b>	85.12	29.92	67.66	30.60	51.94	73.35
ANMI	95.75	96.92	97.85	<b>100.0</b>	68.04	35.42	<b>69.09</b>	<b>84.31</b>	51.63	77.67
ACC	<b>99.50</b>	<b>97.07</b>	70.97	95.20	<b>90.67</b>	<b>35.61</b>	53.99	<b>84.31</b>	<b>100.0</b>	<b>80.81</b>
Max $C_i$	99.50	97.20	100.0	100.0	97.37	41.57	92.64	88.03	100.0	90.70

highest linear relationship with  $C_i$  (0.6928), followed by the ANMI (0.5980),  $Div_3$  (0.4082) and  $Div_1$  (-0.3979) measures.

Table 5 presents the Consistency index values achieved by the consensus partitions selected by the cluster ensemble selection measures ( $Div_1$ ,  $Div_3$ , ANMI and ACC) for each data set, the maximum  $C_i$  value of all the produced consensus partitions and the average  $C_i$  values for each best consensus partition selection measure. The consensus partitions for  $Div_1$  and  $Div_3$  measures were selected choosing the consensus partition corresponding to the median of their values, as mentioned in [10]. For the ANMI and ACC measures, the best consensus partition was selected to be the one that maximizes the respective measures.

The quality of the consensus partitions selected by ACC measure was in 6 out of 9 data sets superior or equal to the quality of the consensus partitions selected by the other measures, specifically, in Bars (99.50%), Breast Cancer (97.07%), Iris (90.67%), Log Yeast (35.61%), Optdigits (84.31%) and Spiral (100%) data sets. In Cigar data set, the best consensus partition was selected using  $Div_3$  measure (100%), and the same happened in Half Rings data set together with ANMI. In Std Yeast data set, none of the four measures selected a consensus partition with similar quality to the best produced consensus partition (92.64%). The closed selected consensus partition was selected using ANMI (69.09%). Concerning the average quality of the partitions chosen by the four measures, the ACC measure stands out again, achieving 80.81% of accuracy, followed by ANMI with 77.67%. The  $Div_3$  and  $Div_1$  measures obtained the worst performance with 74.54% and 73.35%, respectively.

## 6 Conclusions

With the aim of combining multiple data partitions into a better consensus partition, several approaches to produce the cluster ensemble and several consensus functions have been developed. With this diversity, very different consensus partitions with very dissimilar qualities can be obtained. This diversity of consensus partitions was exemplified using the Evidence Accumulation Clustering and the Weighted Evidence Accumulation Clustering using Subsampling combination approaches. This paper deals

with the question of choose the best consensus partition from a set of consensus partitions, that best fits a given data set. With this purpose, we proposed the Average Cluster Consistency (ACC) measure, based on a new similarity conception between each data partition belonging to the cluster ensemble and a given consensus partition. We compared the performance of the proposed measure with three other measures for cluster ensemble selection, using 9 data sets with arbitrary shaped clusters, well separated and touching clusters, and different cardinality, dimensionality and cluster densities. The experimental results showed that the consensus partitions selected by ACC measure, usually were of better quality in comparison with the consensus partitions selected by other measures used in our experiments. Therefore, we can say that our approach is a good option for selecting a high quality consensus partition from a set of consensus partitions.

**Acknowledgements.** We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250). This work was partially supported by the Portuguese Foundation for Science and Technology (FCT), Portuguese Ministry of Science and Technology, under grant PTDC/EIACCO/103230/2008.

## References

1. Fred, A.L.N.: Finding consistent clusters in data partitions. In: Kittler, J., Roli, F. (eds.) MCS 2001. LNCS, vol. 2096, pp. 309–318. Springer, Heidelberg (2001)
2. Strehl, A., Ghosh, J.: Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.* 3, 583–617 (2003)
3. Fred, A.L.N., Jain, A.K.: Combining multiple clusterings using evidence accumulation. *IEEE Trans. Pattern Anal. Mach. Intell.* 27(6), 835–850 (2005)
4. Duarte, F.J., Fred, A.L.N., Rodrigues, M.F.C., Duarte, J.: Weighted evidence accumulation clustering using subsampling. In: Sixth International Workshop on Pattern Recognition in Information Systems (2006)
5. Fern, X., Brodley, C.: Solving cluster ensemble problems by bipartite graph partitioning. In: ICML 2004: Proceedings of the Twenty-First International Conference on Machine Learning, vol. 36. ACM, New York (2004)
6. Topchy, A.P., Jain, A.K., Punch, W.F.: A mixture model for clustering ensembles. In: Berry, M.W., Dayal, U., Kamath, C., Skillicorn, D.B. (eds.) SDM. SIAM, Philadelphia (2004)
7. Jouve, P., Nicoloyannis, N.: A new method for combining partitions, applications for distributed clustering. In: International Workshop on Paralell and Distributed Machine Learning and Data Mining (ECML/PKDD 2003), pp. 35–46 (2003)
8. Topchy, A., Minaei-Bidgoli, B., Jain, A.K., Punch, W.F.: Adaptive clustering ensembles. In: ICPR 2004: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR 2004), vol. 1, pp. 272–275. IEEE Computer Society, Los Alamitos (2004)
9. Topchy, A., Jain, A.K., Punch, W.: Combining multiple weak clusterings, pp. 331–338 (2003)
10. Hadjitarov, S.T., Kuncheva, L.I., Todorova, L.P.: Moderate diversity for better cluster ensembles. *Inf. Fusion* 7(3), 264–275 (2006)
11. Hubert, L., Arabie, P.: Comparing partitions. *Journal of Classification* (October 1985)
12. Kuncheva, L., Hadjitarov, S.: Using diversity in cluster ensembles, vol. 2, pp. 1214–1219 (October 2004)

13. Duarte, F., Duarte, J., Fred, A., Rodrigues, F.: Cluster ensemble selection - using average cluster consistency. In: International Conference on Discovery and Information Retrieval (KDIR 2009), Funchal, October 6-8, pp. 85–95 (2009)
14. Sneath, P., Sokal, R.: Numerical taxonomy. Freeman, London (1973)
15. King, B.: Step-wise clustering procedures. *Journal of the American Statistical Association* (69), 86–101 (1973)
16. MacQueen, J.B.: Some methods of classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297 (1967)
17. Ng, R.T., Han, J.: Clarans: A method for clustering objects for spatial data mining. *IEEE Trans. on Knowl. and Data Eng.* 14(5), 1003–1016 (2002)
18. Karypis, G., Han, E., News, V.K.: Chameleon: Hierarchical clustering using dynamic modeling. *Computer* 32(8), 68–75 (1999)
19. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic subspace clustering of high dimensional data for data mining applications. *SIGMOD Rec.* 27(2), 94–105 (1998)
20. Guha, S., Rastogi, R., Shim, K.: Cure: an efficient clustering algorithm for large databases. In: SIGMOD 1998: Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, pp. 73–84. ACM, New York (1998)
21. Ester, M., Kriegel, H.P., Jörg, S., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise (1996)
22. Wang, W., Yang, J., Muntz, R.R.: Sting: A statistical information grid approach to spatial data mining. In: VLDB 1997: Proceedings of the 23rd International Conference on Very Large Data Bases, pp. 186–195. Morgan Kaufmann Publishers Inc., San Francisco (1997)
23. Ward, J.H.: Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* 58(301), 236–244 (1963)

**PART II**

**Knowledge Engineering and**

**Ontology Development**

# Cross-Lingual Evaluation of Ontologies with Rudify

Amanda Hicks<sup>1</sup> and Axel Herold<sup>2</sup>

<sup>1</sup> University at Buffalo, 135 Park Hall, Buffalo, NY 14260, U.S.A.

<sup>2</sup> Berlin-Brandenburgische Akademie der Wissenschaften  
Jägerstr. 22/23, 10117 Berlin, Germany

**Abstract.** Rudify is a set of tools used for automatically annotating concepts in an ontology with the ontological meta-properties employed by OntoClean [1]. While OntoClean provides a methodology for evaluating ontological hierarchies based on ontological meta-properties of the concepts in the hierarchy, it does not provide a method for determining the meta-properties of a given concept within an ontology. Rudify has been developed to help bridge this gap, and has been used in the KYOTO project to facilitate ontology development. The general idea behind Rudify is the assumption that a preferred set of linguistic expressions is used when talking about ontological meta-properties. Thus, one can deduce a concept's meta-properties from the usage of the concept's lexical representation (LR) in natural language. This paper describes the theory behind Rudify, the development of Rudify, and evaluates Rudify's output for the rigidity of base concepts in English, Dutch, and Spanish. Our overall conclusion is that the decisive output for English is useable data, while the procedure currently exploited by Rudify does not easily carry over to Spanish and Dutch.

**Keywords:** Ontology, Ontology evaluation, Rigidity, Essence, Base concept, Wordnet, KYOTO.

## 1 Introduction

Developing an ontology requires paying especial attention to the hierarchical relations. In particular, taking into consideration certain meta-properties of the concepts modelled in the ontology can help the developer avoid formal contradiction and unsound inheritance of properties [1]. However, manually determining ontological meta-properties of concepts within large ontologies is time consuming and has been shown to produce a low level of agreement amongst human annotators [10]. A further difficulty around the annotation of meta-properties is that evaluating the meta-properties of concepts can be difficult for non-ontologists while evaluating technical concepts from a specific domain may be difficult for ontologists who are not trained in this domain.

In this paper we present Rudify, a set of tools for the automatic determination of ontological meta-properties. Rudify has been used for ontology development within the KYOTO project [15,16].

Section 2 of this paper provides an overview of the KYOTO project with particular emphasis on the role of the ontology. Section 3 contains a brief description of

OntoClean, a method for evaluating hierarchical relations in an ontology [3]. Section 4 discusses the meta-property of rigidity and its relation to the type–role distinction. Section 5 discusses the development of Rudify. In Sect. 6 the notion of base concepts is briefly introduced. A set of base concepts was used for the evaluation of the Rudify output (Sect. 7). Finally, Sect. 8 provides specific examples of how Rudify output can be used to “clean up” hierarchical relations within an ontology.

## 2 The KYOTO Project

The KYOTO project is a content enabling system that performs deep semantic analysis and searches and that models and shares knowledge across different domains and different language communities. Semantic processors are used for concept and data extraction, and the resulting knowledge can be used across the different linguistic communities. A wiki environment allows domain specialists to maintain the system. KYOTO is currently being targeted toward the environmental domain and will initially accommodate seven languages, namely, English, Dutch, Spanish, Italian, Basque, Chinese, and Japanese. The system depends on an ontology that has been linked to lexical databases (wordnets) for these languages. The role of the ontology is to provide a coherent, stable and unified frame of reference for the interpretation of concepts used in automatic inference. For more information on the KYOTO project see [14] and <http://www.kyoto-project.eu/>.

KYOTO should be able to accommodate not only a variety of languages and domains of knowledge, but also *changes* in scientific theories as both the world and our knowledge of the world change. We, therefore, require an ontology that is not idiosyncratic, but rather one that can accommodate

1. a variety of languages and their wordnets,
2. a variety of scientific domains,
3. a variety of research communities,
4. future research in these domains, and
5. can serve as the basis of sound, formal reasoning.

Because the end users will be able to maintain and extend the ontology, it is crucial that the ontology is extended in a clean and consistent manner by non-ontology experts.

With this aim in mind we have developed Rudify. We are using Rudify in conjunction with OntoClean in order to build and maintain a clean ontology. By evaluating the ontological meta-properties of concepts, Rudify facilitates a major step in the construction and maintenance of clean hierarchies.

## 3 OntoClean

OntoClean [3] is a method for evaluating ontological taxonomies. It is based on ontological meta-properties of the concepts that appear in the ontological hierarchy. These meta-properties – namely, rigidity, unity, identity, and dependence – are both highly general and based on philosophical notions. Although OntoClean uses meta-properties

to evaluate ontological taxonomies, it is not intended to provide a way of determining the meta-properties themselves. Instead it shows the logical consequences of the users modelling choices, most notably ontological errors that may result in taxonomies after modelling choices have been made [1]. Rudify helps fill this gap by assigning meta-properties to concepts based on how the concepts are expressed in natural language.

Of the four types of ontological meta-properties used by OntoClean, we focus on rigidity. There are several reasons for this choice. First – and most important in the context of the KYOTO project, the notion of rigidity plays a large role in the distinction between types and roles, since every type is a rigid concept and every role is a non-rigid concept. Second, it is relatively easy to find lexical patterns for rigidity. The lexical patterns are a crucial prerequisite for the programmatical determination of meta-properties as done by Rudify (see Sect. 5). Third, AEON [11] also concentrated on rigidity, so there is a basis for comparison of data.

## 4 Rigidity

The notion of rigidity relies on the philosophical notion of essence. An essential concept is one that necessarily holds for all of its instances. For example, being an animal is essential to being a cat since it is impossible for a cat to not be an animal, while being a pet is not essential because any cat can, in theory, roam the streets and, thereby, not be a pet. The idea of essence contains an idea of permanence; Fluffy the cat is an animal for the entire duration of his life. However, the notion of essence is stronger than permanence. While Fluffy can be a pet for his entire life, it nevertheless remains possible for him to cease being a pet.

Armed with the notion of essence, we can now define rigidity. A rigid concept is a concept that is essential to all of its possible instances, i. e., every thing that *could* be a cat *is* in fact a cat. Therefore, “cat” is a rigid concept. However, “pet” is a non-rigid concept since there are individual pets that do not have to be a pet.

Non-rigidity further subdivides into two meta-properties: semi-rigidity and anti-rigidity. Those concepts that are essential to some, but not all, of their instances are semi-rigid, while those that are not essential to any of their instances are anti-rigid. We do not focus on this distinction in our work although Rudify can be used to evaluate these meta-properties as well.

We are currently using Rudify to develop the central ontology for the KYOTO project and to separate type- and role-hierarchies therein. This section provides a discussion of the relation between rigidity and type–role hierarchies.

Types and roles are the two main subdivisions of sortal concepts. A sortal concept is a concept that describes what sort of thing an entity is. For example “cat,” “hurricane,” and “milk” are sortal concepts while “red,” “heavy,” and “singing” are not. In an ontology, sortal concepts are those concepts that carry the meta-property identity (for a discussion of identity, see [1]). Furthermore, sortals usually correspond to nouns in natural language. We work on the assumption that the concepts represented in the noun hierarchy of WordNet [5, see also Sect. 6] are sortal terms, since this is generally the case. Types are rigid sortals, while non-rigid sortals are generally roles. Furthermore, roles cannot subsume types.



**Fig. 1.** Erroneous hierarchy below “cat”

In order to see that roles should not subsume types, we can consider the (erroneous) hierarchy in Fig. 1. According to this hierarchy, if Fluffy ceases to be a pet, then Fluffy also ceases to be a cat, which is impossible.

From this last point in conjunction with the above assumption that nouns usually represent sortals, it follows from the OntoClean principles that amongst sortal terms, non-rigid sortals should not subsume rigid sortals. In other words, non-rigid nouns generally should not subsume rigid nouns. There are exceptions to this rule. However, this general conclusion allows us to evaluate concepts only for rigidity and non-rigidity, which in turns saves us the computationally expensive task of evaluating non-rigid terms as either semi- or anti-rigid over large sets of concepts.

## 5 Rudify Development

The general idea behind Rudify is the assumption that a preferred set of linguistic expressions is used when talking about ontological meta-properties. Thus, one can deduce a concept’s meta-properties from the usage of the concept’s lexical representation (LR) in natural language. This idea has been developed and programmatically exploited first in the AEON project [11]. AEON was developed for the automatic tagging of existing ontologies in terms of OntoClean meta-properties. The KYOTO project decided to rewrite the software based on the principles published in [10] for several reasons: there was no active development of the tool any more and the software was released as a development snapshot only, the web service interface had to be changed due to the maintenance stop of the originally implemented one by Google, and a more flexible input facility was needed instead of the purely OWL based one.

In the following technical description and discussion of Rudify we focus on the meta-property of rigidity as this has been the most important property in the context of the KYOTO project so far.

The first step in the Rudify process is the identification of adequate LRs for the concepts that are to be tagged. Due to polysemous word forms there is no one to one mapping between concepts and LRs. Also, the actual number of recorded senses for a given LR may vary across lexical databases and across versions of a specific lexical database. The results reported here are based on the English WordNet [5] version 3.0, and on the Spanish and Dutch wordnets. A further complication are concepts that do not have LRs at all. Typically, this applies mostly for concepts of the top levels of ontologies, although there are some (rare) examples like the missing English antonym for “thirsty” meaning “not thirsty” which constitutes a lexical gap.

A set of lexical patterns that represent positive or negative evidence for a single meta-property needs to be developed. Each pattern specifies a fixed sequence of word forms. For little inflecting languages like English with relatively fixed word order this approach

works reasonably well. Further refinement of the patterns is needed for languages with more free word ordering. For rigidity, we found only patterns representing evidence *against* rigidity. Thus, the default assumption when tagging for rigidity is that rigidity applies. A concept  $C$  is considered non-rigid only if enough evidence against rigidity has been collected for  $C$ . Obviously, sparse data for occurrences the LR for  $C$  will distort the results and produce a skew in the direction of rigidity.

For rigidity, a typical pattern reads “would make a good  $X$ ” where  $X$  is a slot for a concept’s LR. This may be a single token, a multiword or even a complex syntactic phrase (as is frequently the case in e.g. Romance languages). Over-generation of patterns is prevented by enumerating and excluding extended patterns. The non-rigid pattern “is no longer (—/a/an)  $X$ ” over-generates phrases like “there is no longer a *cat* (in the yard/that could catch mice/...)” from which we cannot deduce non-rigidity for “*cat*.”

Another frequent over-generation is found for LRs that occur as part of a more complex compound noun as in “is no longer an *animal shelter*” where *animal* is not an instance of the concept “animal.” As the results returned from web search engines are often mere fragments of sentences such instances can only be excluded based on part-of-speech tagging and not reliably based on (chunk) parsing.

Rudify currently uses 25 different lexical patterns for English as evidence against rigidity. The results of the web search queries based on these patterns form a feature vector for each LR that is then used for classification, i. e. the mapping from the feature vector to the appropriate rigidity tag. Technically this is a ternary decision between *rigid*, *non-rigid* and *uncertain*.

All classifiers were trained on a hand crafted and hand tagged list of 100 prototypical LRs of which 50 denote rigid concepts and 50 denote non-rigid concepts. They cover a broad range of domains and are recorded as monosemeous (having a single sense) in the wordnets they were taken from.

Four different algorithms have been used for classification:

- decision tree (J48, an implementation of C4.5)
- multinomial logistic regression
- nearest neighbor with generalization (NNge)
- locally weighted learning, instance based

In evaluating the output we considered the results of all four classifiers and ranked the results according the degree of consensus amongst them (see Sect. 6 for more details).

Both Rudify and AEON rely on the World Wide Web as indexed by Google as the hugest repository of utterances that is accessible to the research community. This is done in order to minimize sparse data effects. We are aware of the theoretical implications that data extracted from Google or other commercial web search engines entails. The most crucial problems are:

- Results are unstable over time. The indexing process is rerun regularly and results retrieved at any given point in time may not be exactly reproducible later.
- The query syntax may be unstable over time and implements boolean rather than linguistically motivated searches.
- There are arbitrary limitations of the maximum number of results returned and of the meta-data associated with each result. These may also change over time.

- The data repository is in principle uncontrolled as write access to the World Wide Web and other parts of the Internet is largely unrestricted. Commercial search engines work as additional filters on the raw data with their filter policy often left undocumented and subject to changes as well.

From a linguist's point of view, the first three of these problems are discussed in more detail in [9].

Rudify now is a highly configurable modular tool with parameter sets developed for English, Dutch, Italian, Spanish, Basque and Japanese. The software is written in Python and NLTK [13] is used as the linguistic backend. Classifier creation, training and application is done using Weka 3 [12], but can be easily delegated to any software suite capable of manipulating ARFF files. Rudify is released as free and open source software at <http://rudify.sourceforge.net/>.

## 6 Base Concepts

In [6], the presence of basic level concepts (BLC) in human cognition is empirically demonstrated. In a conceptual taxonomy, for each concept  $C$  its subordinate concepts  $C_n$  are typically more specific than  $C$ . The increase in specificity is due to at least one added feature for  $C_n$  that is compatible with  $C$  but allows for discrimination between all  $C_n$ . BLCs mark the border between the most general concepts comprising only few features and the most feature rich concepts.

Base concepts (BC) as described in [8] are those concepts within a semantically structured lexical data base that "play the most important role" in that data base. This intuitive but vague notion is effectively a rephrase of the BLC. BCs, though, are conceived as a purely computationally derived set based on semantic relations encoded in hierarchical lexical databases. BCs are those concepts that are returned by the following algorithm: for each path  $p$  from a leaf node (a node with no hyponym relation to other nodes) up to a root node (a node with no hypernym relation to other nodes) choose the first node  $C$  with a local maximum of specific relations to other nodes as a BC. This algorithm can be adapted by constraining the set of specific relations (e. g. considering only hyponymy or accepting all encoded relations including lexical relations) and by defining a minimally required number of concepts a possible BC must be related to. BC sets depend on these specified parameters and the hierarchical structure of the lexical database. Thus, different sets are computed for different versions of WordNet and other national wordnets. Software and data for computing BCs from wordnets in the WordNet format are freely available online at <http://adimen.si.ehu.es/web/BLC>.

WordNet [5] is an electronic lexical database for English. It organizes words in terms of semantic relations including synonymy ("car"—"automobile"), hyponymy (the relation among general and specific concepts, like "animal" and "cat," that results in hierarchical structures), and meronymy (the part-whole relation, as between "cat" and "claw"). Linking words via such relations results in a huge semantic network.

We have added a set of BCs to the middle level of the KYOTO ontology thereby providing the ontology with a generic set of concepts that can be used for inter-wordnet mappings and wordnet to ontology mappings.

Rudify was initially evaluated on the set of BCs derived from WordNet 3.0 considering only hypernym relations and with a minimum of 50 subsumed concepts for each BC. These parameters result in a set of 297 concepts (BC-50-ENG). Inspecting the BC-50-ENG set we found LRs that highly unlikely denote BLCs though they fulfill the formal criteria for BCs. A striking example is “moth.” In WordNet, much effort was spent to record a high number of different insects as distinguished concepts thus effectively shifting the basic level downwards in the taxonomic tree. A similar effect of basic level shifts can be shown for experts in their respective domain [7].

## 7 Evaluation of Output

We tested Rudify on four different English language data sets:

- 50 region terms (handcrafted by environmental domain specialist)
- 236 Latin species names (selected by environmental domain specialist)
- 201 common species names (selected by environmental domain specialist)
- 297 basic level concepts (BC-50-ENG).

We then repeated the test for other European languages:<sup>1</sup>

- 266 basic level concepts for Spanish (BC-50-ESP)
- 179 basic level concepts for Dutch (BC-50-NLD)

### 7.1 Domain Specific Terms (English)

Classifiers correctly classified all region terms and all Latin species names as rigid concepts. This holds also for the common English species names with three exceptions: “wildcat” was misclassified as denoting a non-rigid concept by all four classifiers and “wolf” and “apollo” (a butterfly) were mis-classified by all classifiers except NNge. This mis-classification is due to the fact that those LRs are not monosemously denoting a single concept (a species) but are polysemous and also frequently used in figurative language (examples are taken from our log files):

- “Mount Si High School teacher Kit McCormick is no longer a Wildcat.”  
(generalization from a school mascot to a school member)
- “Also the 400 CORBON is no longer a wildcat.”  
(a handgun)
- “He nearly gave in and became a Wildcat before finally deciding to honor his original commitment to the Ducks.”  
(a football team’s (nick)name)
- “For example, the dog is no longer a wolf, and is now a whole separate species.”  
(example discusses changing relations between concepts over time)

---

<sup>1</sup> Montse Cuadros and Eneko Agirre provided important help with developing training sets, lexical patterns and the BC-50 set for Spanish; Roxane Segers and Wauter Bosma kindly helped us with the Dutch data.

- “For four years, the space agency had been planning, defining, or defending some facet of what led up to and became Apollo.”  
(a space mission’s name)
- “Others figuring prominently in the county’s history were Edward Warren, who established a trading post near what is now Apollo [...]”  
(a geographical name)
- “The patron of the city is now Apollo, god of light, [...]”  
(a Greek deity)

## 7.2 BC-50-ENG

We classify the Rudify output on the BC-50-ENG set according to the agreement amongst the four classifiers used. We refer to those cases in which all four classifiers reached agreement as *decisive*. Rudify yielded decisive output for 215 BCs. Whenever there is disagreement amongst the classifiers, we refer to this output as *difficult*. There are 82 difficult cases that subdivide into two further cases. When three out of four classifiers reached agreement, we refer to this output as *indecisive*. Rudify yielded indecisive output for 56 BCs. When two classifiers evaluate a term as rigid and two as non-rigid, we refer to this as *undecided*. Rudify is undecided with respect to 26 BCs. These figures are summarized in Tab. 1.

**Table 1.** General overview of the classification on the BC-50-ENG set

Rudify output	number of cases
decisive	215
difficult	82
difficult: indecisive	56
difficult: undecided	26

An evaluation of Rudify output for the 215 decisive cases indicates that Rudify produces a high level of accuracy for decisive cases (see Tab. 2). 85 % of the terms evaluated as rigid were correctly evaluated, and 75 % of the terms evaluated as non-rigid are correctly evaluated. Many of the Rudify errors either came from high level concepts, e. g., “artifact” and “unit of measurement,” which are ordinarily dealt with manually, or else they dealt with polysemous words, which was an anticipated difficulty (see Sect. 5).

In 3 % of the decisive output we used Rudify to determine whether a concept is rigid or non-rigid, e. g. for “furniture.” Since not every concept is ontologically clear cut, and since some concepts lie within areas of ontology in which the alternative theories have not yet been properly worked out (e. g., the ontology of artefacts), we have determined that Rudify can be occasionally helpful in making modelling choices based on the common sense uses of the concepts in language. For these cases the evaluation remains unclear.

For 56 concepts Rudify yielded indecisive output. Exactly 50 % of these cases are incorrect (28 out of 56). For this reason we do not regard the indecisive output to be usable data.

The decisive Rudify output on the BCs within the underlying WordNet hierarchy yields five OntoClean errors if we count the hypernyms, and 22 errors if we count instances of hypernym relations. This is based only on the Rudify output prior to evaluating the correctness of this output, but it gives us an idea of the OntoClean results if we uncritically use Rudify to evaluate concepts in the ontology (for more details, see [16]). In short, Rudify output coupled with the OntoClean methodology provides a useful tool for drawing attention to problems in the backbone hierarchy.

In summary, our evaluation of Rudify output on English BCs is that Rudify is successful with respect to the decisive output. It produces decisive output with a relatively high degree of accuracy (83 %) and an overall accuracy on the BC-50-ENG set of 69 % (Tab. 3). Furthermore, Rudify has also proven useful in deciding how to model a few concepts.

**Table 2.** Overview of the decisively classified BC-50-ENG concepts (215 concepts)

class	evaluation	number of cases
rigid	incorrect	20 (12 %)
	correct	142 (85 %)
	unclear	5 (3 %)
non-rigid	incorrect	12 (25 %)
	correct	36 (75 %)

**Table 3.** Summary of evaluation for the BC-50-ENG set

classification	number of cases	
correct	206	(69 %)
incorrect	60	(20 %)
undecided	26	(9 %)
decision left to Rudify	5	(2 %)

### 7.3 BC-50-ESP and BC-50-NLD

The BC-50-ESP and BC-50-NLD sets were computed based on the same principles as described in Sect. 6 on the basis of the Spanish and Dutch wordnets.

For both BC-50-ESP and BC-50-NLD there is significantly less agreement between the classifiers than for BC-50-ENG. Only about a third of the concepts for Spanish and Dutch are decisively classified (see Tab. 4) whereas more than 70 % of the English data was decisively classified.

Within the set of decisively classified BCs agreement with human judgment was also considerably lower for Dutch and Spanish than for the English data set (see Tab. 5).

For three Spanish BCs the decision was left to Rudify: “americano” (‘American’), “asiático” (‘Asian’), and “europeo” (‘European’). The human annotators explicitly marked these BCs as difficult to evaluate. This is due to the fact that they denote strongly culturally motivated concepts for which a universally valid decision cannot be provided.

**Table 4.** General overview of the classification on the BC-50-ESP und BC-50-NLD sets

Rudify output	number of cases	
	BC-50-ESP	BC-50-NLD
decisive	94	56
difficult	172	123
difficult: indecisive	103	60
difficult: undecided	69	63

**Table 5.** Overview of the decisively classified BC-50-ESP and BC-50-NLD concepts

class	evaluation	number of cases	
		BC-50-ESP	BC-50-NLD
rigid	incorrect	29 (45 %)	9 (20 %)
	correct	33 (52 %)	36 (80 %)
	unclear	2 (3 %)	—
non-rigid	incorrect	12 (40 %)	9 (90 %)
	correct	18 (60 %)	1 (10 %)

**Table 6.** Summary of evaluation for the BC-50-ESP and BC-50-NLD sets

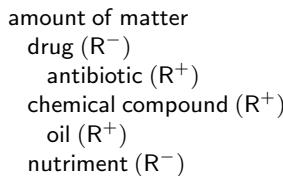
classification	number of cases	
	BC-50-ESP	BC-50-NLD
correct	112 (42 %)	69 (39 %)
incorrect	83 (31 %)	44 (25 %)
undecided	68 (26 %)	63 (35 %)
decision left to Rudify	3 (1 %)	3 (2 %)

While the human annotators eventually tagged these concepts as rigid, Rudify decisively tagged “americano” and “europeo” as non-rigid and was undecided on “asiático”. In the BC-50-NLD set we also find concepts that denote regional affiliation, namely “Amerikaan” (‘American’), “Aziaat” (‘Asian’), “Indiaan” (‘native American’) that provoke varying judgements among the human annotators. These Dutch BCs are judged as rigid by Rudify.

The positive evaluation of Rudify’s performance on English data cannot be confirmed for Spanish and Dutch. The overall accuracy for Spanish and Dutch BCs is at about 40 % and on a large fraction of the data – more than a quarter – the classifiers were undecided (see Tab. 6).

## 8 Application of Output

In this section we illustrate with two English examples how Rudify results can be used to inform ontology design. The first example uses Rudify independently, the second uses Rudify in conjunction with OntoClean principles.



**Fig. 2.** Provisional hierarchy for “amount of matter” taken from WordNet.  $R^+$  indicates a rigid concept,  $R^-$  indicates a non-rigid concept.

### Example 1

We consider BCs that can reasonably be considered amounts of matter. Amounts of matter are generally referred to by mass nouns; ‘milk,’ ‘mud,’ and ‘beer’ are a few examples. Once again we begin by provisionally modelling the concepts taken from WordNet as the upper level concept “amount of matter” into the hierarchy in Fig. 2, which includes rigidity assignments from Rudify.

Using the Rudify data, we can clean up this hierarchy. First we notice that Rudify has evaluated “nutriment” as non-rigid. This indicates that it is probably a role rather than a type. In order to verify this, we refer to the definition taken from WordNet: “a source of materials to nourish the body.” That is, the milk in my refrigerator is a nutriment only if it nourishes a body. If you bathe in milk, like Cleopatra, it is a cosmetic. “Nutriment,” therefore, is a role that milk can play, so it does not belong in the type hierarchy. We therefore, move it to the role hierarchy as subclass of “amount of matter role.” We pause to notice that in this case, the decision was made using Rudify results and human verification of the output. This case does not invoke OntoClean, i. e., there would be no OntoClean errors if “nutriment” were subsumed by “amount of matter.” This contrasts with the second example, which yields a formal error within the hierarchy itself.

### Example 2

Notice that Rudify evaluates “drug” as non-rigid, and “antibiotic” as rigid. However, the current hierarchy in Fig. 2 subsumes the rigid concept under the non-rigid concept. This results in a formal error in the hierarchy. Because “drug” and “antibiotic” are both sortal terms, this means a role subsumes a type, which, as we have seen above leads to inconsistency. Consider the antibiotic penicillin. Penicillin is only a drug if it is administered to a patient, but it is always an antibiotic due to its molecular structure. By subsuming “antibiotic” under “drug,” the ontology erroneously states that if some amount of penicillin is not administered to a patient, then it is not an antibiotic. The solution then, is to move “drug” out of the type hierarchy and into the role hierarchy. “Drug” then becomes a “substance role,” and an antibiotic is subclass of “amount of matter” that can play the role “drug.”

Because “chemical compound” and “oil” are both evaluated as rigid we do not need make any changes to this part of the ontology.

The result is the hierarchy fragments under “amount of matter” and “amount of matter role” in Fig. 3.

amount of matter	amount of matter role
antibiotic	drug
chemical compound	nutriment
oil	

**Fig. 3.** Consistent hierarchies for “amount of matter” and “amount of matter role”

## 9 Conclusions

We presented Rudify – a system for automatically deriving ontological meta-properties from large collections of text based on the lexical representation of individual concepts in natural language. This approach yields valueable results for use in consistency checking of general large scale ontologies such as the KYOTO core ontology. On the basis of 297 basic concepts derived from the English WordNet 69 % agreement with human judgement could be demonstrated. This closely matches the figures reported in [11] for human inter-annotator agreement. For specialized domain terms, agreement was substantially higher: only 3 out of 201 English species terms had been mis-classified.

The evaluation of the results reported here shows potential for further improvement. Word sense disambiguation will increase the accuracy for polysemous words. First experiments involving hypernyms of LRs in the retrieval of evidence for or against ontological meta-properties give already promising results. For any given concept its LRs are not necessarily polysemous in different languages. This fact may be helpful in dealing with the problem of polysemy if data from different languages are combined. The underlying corpus is bigger if several languages are considered and thus there is a greater chance for discovering occurrences of the lexical patterns.

However, the Spanish and Dutch data show that the procedure exploited by Rudify does not carry over to other languages easily. The main reason is the size of the corpus – there is probably much more English text indexed by Google than text in either Spanish and Dutch.<sup>2</sup>

Further elaboration on the lexical patterns will improve the accuracy especially for languages with a more free word order than the English one that allow for more variable patterns.

For future reference and reproducability of the results it will be beneficial to use a controlled linguistic corpus of appropriate size instead of commercial web search engines.

**Acknowledgements.** The development of Rudify and its application to the KYOTO core ontology has been carried out in the EU’s 7th framework project *Knowledge Yielding Ontologies for Transition-based Organizations* (KYOTO, grant agreement no. 211423).

The authors would like to thank Christiane Fellbaum for many fruitful discussions and the KYOTO members for their kind collaboration.

<sup>2</sup> There are no exact figures available from Google, though. An estimate based on the avarage number of total hits for the BCs shows that there is probably four times as much English text as Spanish and five times as much as Dutch.

## References

1. Guarino, N., Welty, C.: An Overview of OntoClean. In: Staab, S., Studer, R. (eds.) *Handbook on Ontologies*, pp. 151–172. Springer, Berlin (2004)
2. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A.: Sweetening WordNet with Dolce. *AI Magazine* 24(3), 13–24 (2003)
3. Guarino, N., Welty, C.: Evaluating Ontological Decisions with OntoClean. *Communications of the ACM* 45(2), 61–65 (2002)
4. Welty, C., Guarino, N.: Supporting Ontological Analysis of Taxonomic Relationships. *Data and Knowledge Engineering* 39(1), 51–74 (2001)
5. Fellbaum, C. (ed.): *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge (1998)
6. Rosch, E., Mervis, C.B., Gray, W.D., Johnson, D.M., Boyes-Braem, P.: Basic Objects in Natural Categories. *Cognitive Psychology* 8, 382–439 (1976)
7. Tanaka, J.W., Taylor, M.: Object Categories and Expertise: Is the Basic Level in the Eye of the Beholder? *Cognitive Psychology* 23, 457–482 (1991)
8. Izquierdo, R., Suárez, A., Rigau, G.: Exploring the Automatic Selection of Basic Level Concepts. In: *Proceedings of the International Conference on Recent Advances on Natural Language Processing (RANLP 2007)* (2007)
9. Kilgarriff, A.: Googleology is Bad Science. *Computational Linguistics* 33, 147–151 (2007)
10. Völker, J., Vrandecic, D., Sure, Y.: Automatic Evaluation of Ontologies (AEON). In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) *ISWC 2005. LNCS*, vol. 3729, pp. 716–731. Springer, Heidelberg (2005)
11. Völker, J., Vrandecic, D., Sure, Y., Hotho, A.: AEON – An Approach to the Automatic Evaluation of Ontologies. *Applied Ontology* 3(1-2), 41–62 (2008)
12. Witten, I.H., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd edn. Morgan Kaufmann, San Francisco (2005)
13. Bird, S., Klein, E., Loper, E.: *Natural Language Processing with Python*. O'Reilly, Sebastopol (2009)
14. Vossen, P., Agirre, E., Calzolari, N., Fellbaum, C., Hsieh, S., Huang, C., Isahara, H., Kanzaki, K., Marchetti, A., Monachini, M., Neri, F., Raffaelli, R., Rigau, G., Tesconi, M.: KYOTO: A System for Mining, Structuring and Distributing Knowledge Across Languages and Cultures. In: *Proceedings of LREC 2008* (2008)
15. Herold, A., Hicks, A., Rigau, G.: Central Ontology Version 1. Deliverable D6.2, KYOTO Project (2009)
16. Herold, A., Hicks, A., Segers, R., Vossen, P., Rigau, G., Agirre, E., Laparra, E., Monachini, M., Toral, A., Soria, C.: Wordnets Mapped to Central Ontology Version 1. Deliverable D6.3, KYOTO Project (2009)

# Towards a Formalization of Ontology Relations in the Context of Ontology Repositories

Carlo Allocca, Mathieu d'Aquin, and Enrico Motta\*

Knowledge Media Institute (KMi), The Open University, Walton Hall  
Milton Keynes MK7 6AA, U.K.

{c.allocca,m.daquin,e.motta}@open.ac.uk

**Abstract.** In the context of Semantic Web Search Engines is becoming crucial to study relations between ontologies to improve the ontology selection task. In this paper, we describe DOOR - The Descriptive Ontology of Ontology Relations, to represent, manipulate and reason upon relations between ontologies in large ontology repositories. DOOR represents a first attempt in describing and formalizing ontology relations. In fact, it does not pretend to be a universal standard structure. Rather, It is intended to be a flexible, easily modifiable structure to model ontology relations in the context of ontology repositories. Here, we provide a detailed description of the methodology used to design the DOOR ontology, as well as an overview of its content. We also describe how DOOR is used in a complete framework (called KANNEL) for detecting and managing semantic relations between ontologies in large ontology repositories. Applied in the context of a large collection of automatically crawled ontologies, DOOR and KANNEL provide a starting point for analyzing the underlying structure of the network of ontologies that is the Semantic Web.

## 1 Introduction

Ontologies are the pillars of the Semantic Web (SW) and, as more and more ontologies are made available online, the SW is quickly taking shape. As a result, the research community is becoming more and more aware that ontologies are not isolated artifacts: they are, explicitly or implicitly, related with each other [15]. A number of studies have intended to tackle some of the challenges raised by these ontology relationships, from both theoretical and practical points of view.

At a theoretical level, studies have targeted ontology comparison in order to identify overlaps between ontologies [17]. Approaches have been proposed to find differences between versions of an ontologies [18,16]. According to [13], the ontology versioning problem has been defined as *the ability to handle changes in ontologies by creating and managing different variants of it*. In other words, ontology versioning means that there are multiple variants of an ontology around. The authors of [13] suggested that, ideally, developers should maintain not only the different versions of an ontology, but also some information about the way versions differ and whether or not they are compatible with each other. In [8] ontology integration is defined as the construction of an ontology C that formally specifies the union of the vocabularies of two other ontologies

---

\* This work was funded by the EC IST-FF6-027595 NeOn Project.

A and B. The most interesting case is when A and B commit to the conceptualization of the same domain of interest or of two overlapping domains. In particular, A and B may be related by being *alternative ontologies*, *truly overlapping ontologies*, *equivalent ontologies with vocabulary mismatches*, *overlapping ontologies with disjoint domain*, *homonymically overlapping ontologies*. Finally, in ontology matching, an alignment is a set of correspondences between the entities of two ontologies, therefore relating these two ontologies by mapping their models with each other.

At a practical level, Semantic Web Applications use the SW as a large-scale knowledge source [4]: they achieve their tasks by automatically retrieving and exploiting knowledge from the SW as a whole, using advanced Semantic Web Search Engines (SWSEs) such as WATSON [5]. These SWSEs provide keyword based search mechanisms to locate relevant ontologies for particular applications. As an example, the query “*student*” currently gives 1079 ontologies as a result in WATSON<sup>1</sup> (valid on the 22/04/2009). However, these results are provided as a simple list without making explicit the underlying relations that link ontologies with each other. Indeed, on the first page, at least 2 of the ontologies (<http://www.vistology.com/ont/tests/student1.owl> and <http://www.vistology.com/ont/tests/student2.owl>) represent, apart from their URIs and the base namespaces, exactly the same logical model, expressed in the same ontology language. Another common situation is when an ontology has been translated in different ontology languages. This is the case in the first and second results of the query “*student, university, researcher*” (<http://reliant.teknowledge.com/DAML/Mid-level-ontology.owl> and <http://reliant.teknowledge.com/DAML/Mid-level-ontology.daml>). These two ontologies are obviously two different encodings of the same model. Inspecting the results of WATSON in the same way, it is not hard to find ontologies connected with other, more sophisticated semantic relations: *versioning*, *inclusion*, *similarity*, etc. Leaving implicit these relations in SWSE’s ontology repositories generates additional difficulties in exploiting their results, expecting the users and the applications to find the “right” or “best” ontology to achieve their goal.

Both the theoretical and practical challenges concerning relations between ontologies indicate a need for a general study of these relations, providing a formal base for defining, manipulating and reasoning upon the links that relate ontologies online, explicitly or implicitly. Here, we chose to take an ontological approach to this problem. We design DOOR, a Descriptive Ontology of Ontology Relations that defines ontology relations using ontological primitives and rules. Apart from the ontology itself, the main contributions of this work concern the realization of a methodology to identify and define relations between ontologies, as well as the development of a complete system based on DOOR (KANNEL), providing services for detecting relations, populating DOOR, and formally querying detected and inferred relations in a large ontology repository.

This paper is structured as follows: in Section 2 we continue discussing significant work concerning ontology relations; Section 3 presents the adopted methodology for designing DOOR; Section 4 describes the DOOR ontology; In Section 5 we briefly describe KANNEL and the main role of DOOR in this framework. Finally, Section 6 concludes the paper and sheds the light on interesting future research on ontology relations.

---

<sup>1</sup> <http://watson.kmi.open.ac.uk>

## 2 Related Work

J. Heflin [12] was the first to studied formally some of the different types of links between ontologies, focusing on the crucial problems of versioning and evolution. However, currently, there is no ontology management system that implements his framework. The authors of [15] characterized, at a very abstract level, a number of relations between ontologies such as *sameConceptualization*, *Resemblance*, *Simplification* and *Composition*, without providing formal definitions for them, and without considering the links between these relationships. Several approaches have been focusing on how to compare two different versions of ontologies in order to find the differences. In particular, PROMTDIF [18] compares the structure of ontologies and OWLDiff (<http://semanticweb.org/wiki/OWLDiff>) computes the differences by entailment, checking the two set of axioms. SemVersion [21] compares two ontologies and computes the differences at both the structural and the semantic levels. In addition, many measures exist to compute particular forms of similarity between ontologies [6].

All these studies discuss particular relations separately and are generally based on an abstract, informal definition of the relations they consider. A complete model is necessary to provide a wide overview of existing ontology relations, to clearly establish what are their definitions, formal properties, and how they are connected with each other.

## 3 Methodology for the DOOR Ontology

Building an ontology of relationships between ontologies is a very ambitious task. It requires a deep analysis of the ontologies available online and of the literature, at different levels. Therefore, a reasonably rigorous but nonetheless flexible methodology is needed to identify, describe and formalize ontology relations and their connections, in order to build the DOOR ontology. Here, after defining some important elements that will be used in the rest of the paper, we present the steps involved in the methodology we adopted and briefly detail each of them.

### 3.1 Definitions and Requirements

We consider the following definitions:

**Definition 1 (Ontology).** An ontology is a set of axioms (in the sense of the description logic) over a Vocabulary VOC, where VOC is the set of the primitive terms (named entities) employed in the axioms of the ontology;

**Definition 2 (Ontology Space).** An ontology space OS, is a collection of ontologies.

**Definition 3 (Ontology Relation).** Given an ontology space OS, an Ontology Relation is any binary relation defined over OS.

At the most general level, the design of the DOOR ontology was based on three main sources to identify relevant ontology relations:

1. We analyzed the results of SWSEs (e.g., WATSON) to manually identify existing, implicit relations between ontologies in these results.

2. We considered relations described in the literature, such as the ones already mentioned in the previous sections.
3. We also included existing, explicit relations that are primitives of the OWL ontology language.

Also, ontology relations in the DOOR ontology should reflect the following important features:

- they are general enough to be applied to multiple domains;
- they are sufficiently intuitive to reflect general meaning;
- they are formally defined to be processed automatically by inference engines;

### 3.2 Main Steps of the Methodology

To design DOOR, we considered the methodology described in [7] for selecting general ontological categories and adapted it to the problem of ontology relations. As a result, we divided our approach into a number of steps, as follows:

1. Identifying the top level relations between ontologies, considering our three sources (SWSes, literature and existing OWL primitives). At this stage, the task only consists in coming up with a list of relations that should be relevant, giving us a general overview of the different sections of the ontology. Relations such as *inclusion*, *similarity*, *incompatibility* and *previous version* are identified here.
2. Specifying the identified relations, identifying relevant variants and sub-relations. Here, our three sources of relations are also employed to derive relations at a lower level. We also use a more systematic approach, which consists in looking at ontologies (and so ontology relations) from 5 different dimensions that can characterize them:
  - **The Lexicographic level**, which concerns the comparison of the vocabularies of the ontologies.
  - **The Syntactic level**, which concerns the comparison of the sets of axioms that form the ontologies.
  - **The Structural level**, which concerns the comparison of the graph structure formed by the axioms of the ontologies.
  - **The Semantic level**, which concerns the comparison of the formal models of the ontologies, looking in particular at their logical consequences.
  - **The Temporal level**, which concerns the analysis of the evolution of ontologies in time.

For example, considering the relation of *inclusion* identified in the first step and that led to a property *includedIn* in the ontology, we can specify this relation according to three different dimensions (syntactic, structural and semantic), leading to three variants of inclusion between ontologies (*syntacticallyIncludedIn*, *isHomomorphicTo* and *semanticallyIncludedIn*) that consider the set of axioms, the graph and the formal models of the ontologies respectively. In addition, besides the systematic analysis of this relation according to the dimensions, we include in DOOR particular forms of inclusions derived from existing OWL primitives (e.g., OWL *imports*) and from the literature (e.g., *isAConservativeExtensionOf* [9]). More details about these relations are given in the next section.

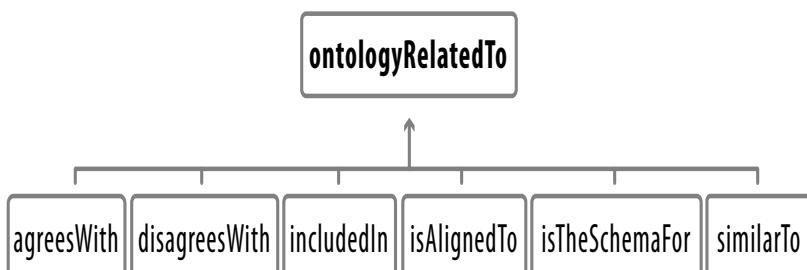
3. Characterizing each relation by its algebraic properties. For example, the algebraic properties for similarity are that it is *reflexive* and *symmetric*. For inclusion, we can define that it is *reflexive* and *transitive*. Including such information in the ontology corresponds to what [7] calls defining the *ground axioms*.
4. Establishing connections between relations. The results obtained from the previous steps are mainly top-level relations with a list of variants, each of them being given algebraic properties. Here, we want to structure these lists, in particular by giving them taxonomic relations. As an example, it can be easily established that *syntacticallySimilarTo* is a sub property of *semanticallySimilarTo*. In the same way, we can indicate that a *previous version* of an ontology ought to be *similar* to it. This corresponds to defining *non-ground axioms* in [7].
5. Introducing rules to define complex relations from atomic ones. For example, the *equivalentTo* property can be defined as *equivalentTo*( $X_1, X_2$ ):- *includedIn*( $X_1, X_2$ ), *includedIn*( $X_2, X_1$ ).

Like in any methodology, the application of these steps should be flexible and continuous. Getting back to a previous step is sometimes necessary and, as the building of an ontology such as DOOR is a constantly ongoing effort, it should be possible to re-apply the methodology entirely to make the ontology evolve.

The intended result is an ontology made, on the one hand, of an ontologically defined and taxonomically structured set of relations, and on the other hand, of a set of rules to define complex relations. In the following we give a detailed overview of the first version of the DOOR ontology, considering only the first (ontological) part of it, as, due to its complexity, the definition of rules governing complex relations is still a work in progress and would not fit in this paper.

## 4 Formal Description of DOOR

The OWL version of the DOOR ontology can be downloaded at: <http://kannel.kmi.open.ac.uk/ontology>. We start with describing the first level of DOOR, in Figure 1. The main relevant abstract relations are simply represented as sub-properties of *ontologyRelatedTo*. An ontology X is *ontologyRelatedTo* another one Y if one of the top level relations is satisfied between X and Y. The top level relations include *includedIn*, *similarTo*, *isAlignedTo*, *disagreesWith*, *agreesWith* and *isTheSchemaFor*. We clustered them in four groups and each group will be explained in more details in the next sub-sections.



**Fig. 1.** Top Level of DOOR

#### 4.1 includedIn and equivalentTo

*includedIn* and *equivalentTo* are two of the main ontology relations. The former represents the meaning of “an ontology contains an another one”. The latter intends to convey the meaning of “two ontologies express the same knowledge”. According to our methodology, these two relations have been analyzed at different levels, giving origin to different kinds of *inclusion* and *equivalence* relations. In Table 1, we summarize the result of these analyses:

**Table 1.** Specialization of inclusion and equivalence relations

	includedIn	equivalentTo
Semantic	semanticallyIncludedIn isAConservativeExtentionOf	semanticallyEquivalentTo
Structural	isHomomorphicTo	isIsomorphicTo
Syntactic	syntacticallyIncludedIn import	syntacticallyEquivalentTo

In particular, the sub-relations of *includedIn* are defined as follows:

***syntacticallyIncludedIn*( $X_1, X_2$ )** if the set of axioms of  $X_1$  is contained in the set of axioms of  $X_2$ , which means  $X_1 \subseteq X_2$ .

***isHomomorphicTo*( $X_1, X_2$ )** if a homomorphism exists between the RDF-graph of  $X_1$  and the RDF-graph of the  $X_2$ .

***semanticallyIncludedIn*( $X_1, X_2$ )** if the set of models of  $X_1$  is contained in the set of models of  $X_2$ . In other words, if  $X_2 \models X_1$ .

***isAConservativeExtentionOf*( $X_1, X_2$ )**, informally, if *syntacticallyIncludedIn*( $X_2, X_1$ ) and all the axioms entailed by  $X_1$  over the vocabulary of  $X_2$  are also entailed by  $X_2$ . A more formal definition can be found in [9]. The notion of conservative extension has been used in particular for ontology modularization [10].

***import*( $X_1, X_2$ )** if there is an explicit statement in  $X_1$  indicating that it imports  $X_2$  using the *owl:imports* primitive. Formally, this means that all the axioms of  $X_2$  should be considered as contained in  $X_1$ .

The sub-relations of *equivalentTo* are defined as follows:

***syntacticallyEquivalentTo*( $X_1, X_2$ )** if and only if *SyntacticallyIncludedIn*( $X_1, X_2$ ) and *SyntacticallyIncludedIn*( $X_2, X_1$ ).

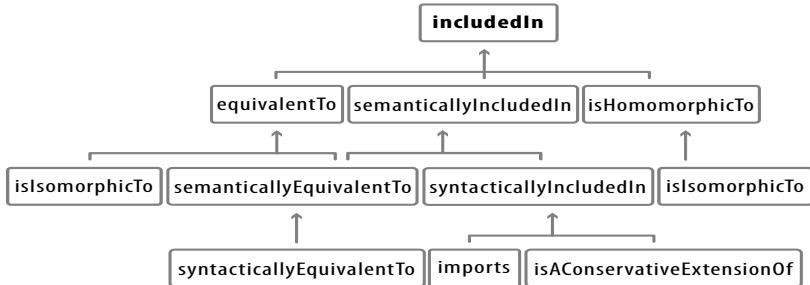
***isIsomorphicTo*( $X_1, X_2$ )** if an isomorphism exists between the graph of  $X_1$  and the graph of  $X_2$ .

***semanticallyEquivalentTo*** if and only if *semanticallyIncludedIn*( $X_1, X_2$ ) and *semanticallyIncludedIn*( $X_2, X_1$ ).

Finally, following our methodology, we defined the algebraic properties of each relation<sup>2</sup> and classified them to create a taxonomic structure relating these relations. This structure is showed in Figure 2<sup>3</sup>.

<sup>2</sup> Since these are fairly obvious, we do not detail them.

<sup>3</sup> The arrows represent the subPropertyOf relation. For example, *syntacticallyEquivalentTo* is a sub property of *semanticallyIncludedIn*.

**Fig. 2.** Taxonomy for includedIn and equivalentTo

#### 4.2 similarTo

Ontology similarity has been described as a measure to assess how close two ontologies are [6]. Various ways to compute the similarity between two ontologies have been described which are relevant in different application contexts. In our work, *similarTo* is used to represent the meaning of “how an ontology overlap/cover parts of the same area as interest of another ontology”. Following our methodology, *similarTo* has been analyzed and formalized at the lexicographic, structural and semantic level, giving origin to different kinds of similarity relations (see Table 2).

**Table 2.** Specialization of the similarity relation

	SimilarTo
Semantic	semanticallySimilarTo MappingSimilarTo
Syntactic	syntacticallySimilarTo
Lexicographic	LexicographicSimilarTo

To define these relations, we need to introduce the following elements: given two ontologies  $X_1$  and  $X_2$ , we denote by  $LC(X_1, X_2)$  the set of axioms of  $X_1$  that are logical consequences of  $X_2$  and by  $Voc(X_1)$  the vocabulary of  $X_1$ . The following definitions depend on a threshold  $T > 0$ .

*semanticallySimilarTo*( $X_1, X_2$ ), if

$$\frac{|LC(X_1, X_2) \cap LC(X_2, X_1)|}{\max(|X_1|, |X_2|)} \geq T$$

*syntacticallySimilarTo*( $X_1, X_2$ ), if

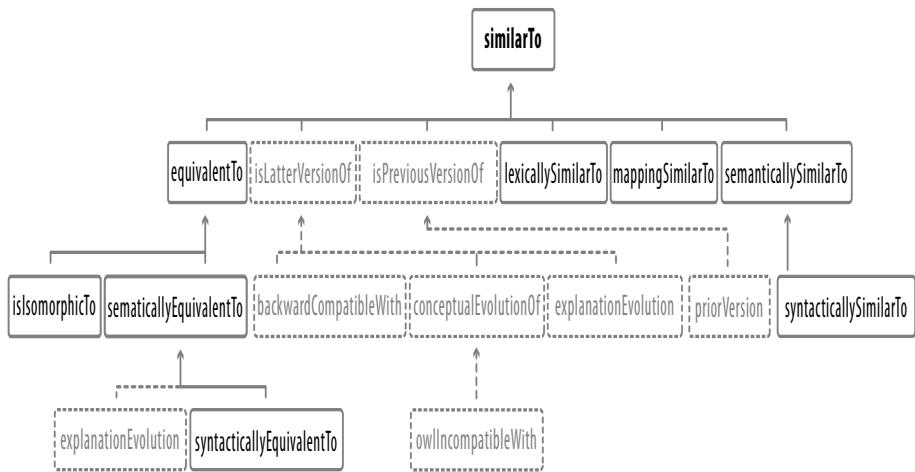
$$\frac{|X_1 \cap X_2|}{\max(|X_1|, |X_2|)} \geq T$$

*lexicographicallySimilarTo*( $X_1, X_2$ ), if

$$\frac{|Voc(X_1) \cap Voc(X_2)|}{\max(|Voc(X_1)|, |Voc(X_2)|)} \geq T$$

Finally, in addition to the relations defined above, we also consider a similarity relation that relies on the existence of an alignment between the two ontologies. Indeed, **mappingSimilarTo** is a relation that links two ontologies  $X_1$  and  $X_2$  if there exists an alignment from  $X_1$  to  $X_2$  and this alignment covers a substantial part of the vocabulary of  $X_1$  (i.e., a proportion greater than a threshold  $T$ ). Note that, since alignments can be unidirectional, *mappingSimilarTo* differs from the other similarity functions by not being symmetric.

Finally, we have classified the relations in Table 1 to create the taxonomic structure showed in Figure 3.



**Fig. 3.** Taxonomy for *similarTo*. Dashed elements represent elements from other sections of the ontology.

### 4.3 Versioning

Versioning is a temporal relation that concerns the evolution of an ontology. In [13], the ontology versioning problem has been defined as *the ability to handle changes in ontologies by creating and managing different variants of it*.

An ontology can evolve over time in different directions, e.g. *lexicographic*, changing the names of some resources, *syntactic*, adding or removing axioms, *semantic*, changing the definition of some concepts or simply adding or removing axioms. Therefore, the new ontology could be equivalent or totally different from the previous one. When we analyze different ways of linking two ontologies by the versioning relation, the two following sentences are suggested immediately: “ $X_1$  is the previous version of the  $X_2$ ” or “ $X_2$  is the latter version of the  $X_1$ ”. These two typical pieces of knowledge are represented in the DOOR ontology by the relations *isPreviousVersionOf* and its inverse *isLatterVersionOf* respectively.

Conforming to our methodology, the *isPreviousVersionOf* and *isLatterVersionOf* relations have been analyzed and formalized, to identify sub-relations and variants. In Table 3 we summarize the result of this analysis.

**Table 3.** Specialization of the versioning relations

	isLatterVersionOf	isPreviousVersionOf
Temporal	conceptualEvolutionOf explanationEvolutionOf backwardCompatibleWith owl:IncompatibleWith	priorVersion
Semantic	conceptualEvolutionOf	
Syntactic	explanationEvolutionOf	

According to [14,12,11] the modification of an ontology can lead to two different types of evolutions: being a conceptual change, meaning that the model of the ontology changed, or being an explanation change, meaning that the modifications happened only at a syntactic level, without affecting the model of the ontology. Therefore, we specialized the *isLatterVersionOf* relation into

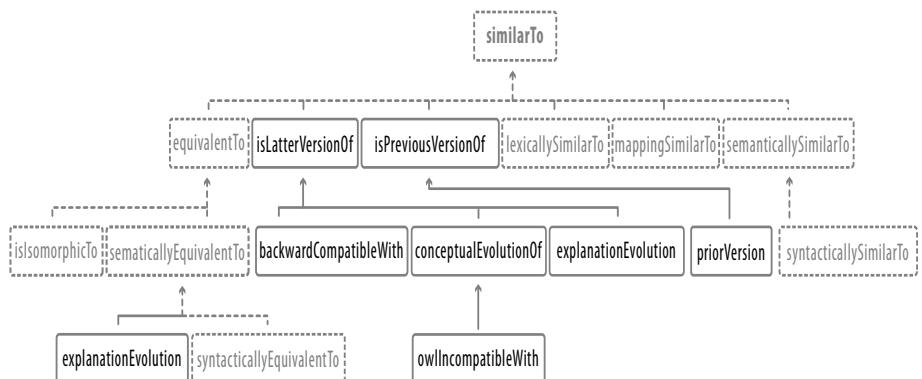
***conceptualEvolutionOf*( $X_1, X_2$ )** if  $X_1$  is a latter version that is not semantically equivalent to  $X_2$ .

***explanationEvolutionOf*( $X_1, X_2$ )** if  $X_1$  is a latter version that is semantically equivalent to  $X_2$

These two relations will lead to the definition of rules to infer them from equivalence and other versioning relations.

In addition, the OWL ontology properties *priorVersion*, *backwardCompatibleWith* and *incompatibleWith* represent explicit relations between versions of ontologies and are included in DOOR as sub-properties of *isLatterVersionOf* and *isPreviousVersionOf*.

To complete this section of the DOOR ontology, we can classified the relations in Table 3 as showed in Figure 4.

**Fig. 4.** Taxonomy for versioning relations

Indeed, according to [14,12,11] the modification of an ontology can lead a new version which is completely different from the original one. But in practice, by analyzing Watson's ontology repository, it is almost always possible establish a similarity between the two ontologies, at least at the lexicographic level. Due to this fact, we chose to consider the versioning relations to be sub-properties of *similarTo*, to indicate that two different versions of the same ontology should, to some extent, be similar. Moreover, in accordance with its definition, the *explanationEvolutionOf* relation is a sub-property of *semanticallyEquivalentTo*.

#### 4.4 Agree and Disagree

Based on the formal measures of the agreement and disagreement between ontologies defined in [3], we introduce the *agreesWith* and *disagreesWith* relations in DOOR. Informally, the former holds the general meaning of “to have the same opinion about something”. In other words, it connects two ontologies, sharing the same knowledge partially and is therefore very related to the *similarTo* and the *equivalentTo* relations. The later indicates that the ontologies “contradict each other” to a certain extent, these contradictions appearing at various levels. Envisaged sub-relations for these two relations are listed in Table 4.

**Table 4.** Specialization of *agreesWith* and *disagreesWith*

	agreeWith	disagreeWith
Temporal	backwardCompatibleWith	owlIncompatibleWith
Semantic	semanticallyEquivalentTo semanticallySimilarTo	hasDisparateModeling incompatibleWith incoherentWith inconsistentWith
Syntactic	syntacticallyEquivalentTo syntacticallySimilarTo explanationEvolution	

In this Table, all the sub-relations of *agreesWith* have already been defined before. We add a few relations to express specific ways for ontologies to disagree, all related to the semantic dimension of the ontologies.

**incompatibleWith( $X_1, X_2$ ):** if *incoherentWith( $X_1, X_2$ )* or *inconsistentWith( $X_1, X_2$ )*.

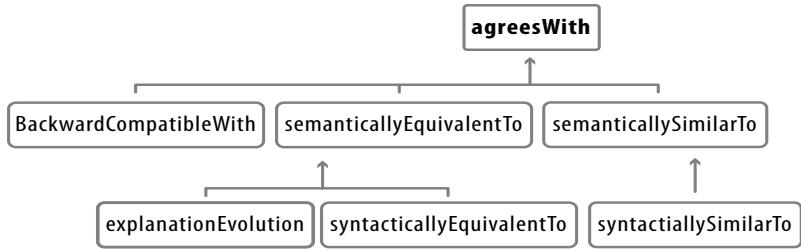
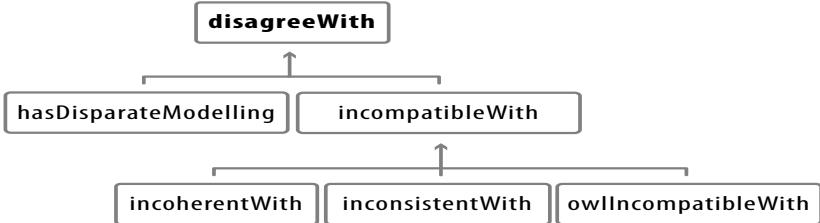
**incoherentWith:** According to [20] an ontology  $X_1$  is incoherent if and only if there is an unsatisfiable concept name in  $X_1$ . Therefore, two ontologies are *incoherent* with each other if, when they are merged, they generate an incoherent result.

**inconsistentWith:** According to [2] an ontology  $X_1$  is inconsistent if it has no model. Therefore, two ontologies are *inconsistent* with each other if, when they are merged, they generate a inconsistent result.

**hasDisparateModeling:** Two ontologies are considered to have disparate modeling if they represent corresponding entities in different ways, e.g. as an instance in one case and a class in the other.

**owl:IncompatibleWith:** It comes from OWL language [19].

Finally, we have also classified the relations in Table 4 as showed in Figures 5 and 6.

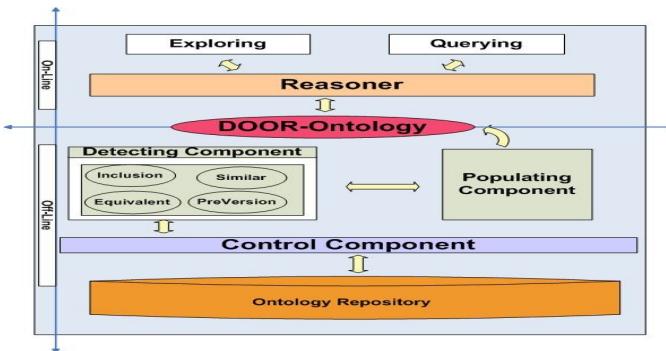
**Fig. 5.** Taxonomy for the agreement relations**Fig. 6.** Taxonomy for the disagreement relations

#### 4.5 Other Relations

Analyzing Watson's ontology repository we found out that there are many documents which only represent the TBox of an ontology and others representing just the ABox. This is captured through the *isTheSchemaFor* relation. *isAlignedTo* relation links ontologies for which exists an alignment.

### 5 Kannel: An Application for the DOOR Ontology

In the previous section, we described the DOOR ontology in detail. Here we provide a brief overview of the way it is used in the KANNEL system. KANNEL [1] is a framework for detecting and managing ontology relations for large ontology repositories, such as WATSON. It is an ontology-based system where the DOOR ontology plays an important role, providing an explicit representation of the implicit relations between ontologies. We have designed an architecture for this framework, as depicted in Figure 7. As showed in this figure, the DOOR Ontology separates the on-line part of the architecture—providing APIs and services that relies on a reasoner—from the off-line part—detecting relations in the repository and populating the ontology. The offline part is based on three components: the *Control Component (CC)*, the *Detecting Component (DC)* and the *Populating Component (PC)*. As a first step, the CC selects from the Ontology Repository ontologies that need to be evaluated to establish potential relations. Then, the selected sets of ontologies are processed by the DC, which contains the main mechanisms to discover the possible relations between ontologies, relying on the definitions provided in this paper. Finally, the PC populates the semantic structure with the



**Fig. 7.** Architecture of the KANNEL framework

detected relations. What is obtained is a set of automatically discovered relations, represented as part of the DOOR ontology so that the reasoner used in the system can infer new relations from the ontological and rule based knowledge included in the ontology. As such, DOOR provides meta-information on the ontology repository in KANNEL.

## 6 Conclusions

In this paper, general relationships between ontologies have been examined. In particular, we have chosen to consider well-known relations in the literature, as well as the ones needed to support the development of Semantic Web Applications. To achieve that, we adapted an ontology building methodology for the construction of DOOR, an ontology of relations between ontologies. This ontology describes relations both from the point of view of their taxonomic structure and from the point of view of their formal definitions, providing the formal properties to describe them as well as a set of rules to derive complex relations from other relations.

We also described KANNEL, a framework for detecting and managing ontology relationships for large ontology repositories. The DOOR ontology plays a fundamental role in KANNEL, not only to provide an explicit representation on ontology relations, but also to supply meta-information that offers several advantages, among which the possibility to reason upon ontologies and their relations. This possibility provides a relevant support for the development of Semantic Web Applications, which can use the semantic web as a large-scale knowledge source [4].

The first version of the DOOR ontology is available in OWL at <http://kannel.kmi.open.ac.uk/ontology>. The KANNEL framework is currently under development. The development of DOOR is obviously a continuous task, which requires a proper assessment of each version. For this reason, we plan to test and validate the first version presented here, in particular by populating it with automatically detected relations between ontologies in WATSON.

## References

- Allocca, C.: Expliciting semantic relations between ontologies in large ontology repositories. PhD Symposium, Poster Session, ESWC (2009)
- Bell, D., Qi, G., Liu, W.: Approaches to inconsistency handling in description-logic based ontologies. In: Chung, S., Herrero, P. (eds.) OTM-WS 2007, Part II. LNCS, vol. 4806, pp. 1303–1311. Springer, Heidelberg (2007)
- d'Aquin, M.: Formally measuring agreement and disagreement in ontologies. In: 5th K-CAP (2009)
- d'Aquin, M., Motta, E., Sabouet, M., et al.: Towards a new generation of semantic web applications. IEEE Intell. Sys. 23(3) (2008)
- d'Aquin, M., Sabou, M., Dzbor, M., Baldassarre, C., Gridinoc, L., Angeletou, S., Mottta, E.: Watson: A gateway for the semantic web. In: Poster Session at 4th ESWC (2007)
- David, J., Euzenat, J.: Comparison between ontology distances (Preliminary results). In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayanan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 245–260. Springer, Heidelberg (2008)
- Gangemi, A., Guarino, N., Masolo, C.: Understanding top-level ontological distinctions (2001)
- Gangemi, A., Pisanello, D.M., Steve, G.: An overview of the onions project: Applying ontologies to the integration of medical terminologies. Technical report. ITBM-CNR, V. Marx 15, 00137, Roma, Italy (1999)
- Ghilardi, S., Lutz, C., Wolter, F.: Did I damage my ontology? a case for conservative extensions in description logics. In: 10th Inter. Conf. (KR), pp. 187–197. AAAI Press, Menlo Park (2006)
- Cuenca Grau, B., Horrocks, I., Kazakov, Y., Sattler, U.: Just the right amount: Extracting modules from ontologies. In: WWW, pp. 717–726. ACM, New York (2007)
- Heflin, J.: Towards the semantic web: Knowledge representation in a dynamic, distributed environment. Ph.D. Thesis, University of Maryland, 2001 (2001)
- Heflin, J., Pan, Z.: A model theoretic semantics for ontology versioning. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 62–76. Springer, Heidelberg (2004)
- Klein, M., Fensel, D.: Ontology versioning on the semantic web. In: Proc. of the Inter. Semantic Web Working Symposium (SWWS), pp. 75–91 (2001)
- Klein, M., Fensel, D., Kiryakov, A., Ognyanov, D.: Ontology versioning and change detection on the web. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) EKAW 2002. LNCS (LNAI), vol. 2473, pp. 197–212. Springer, Heidelberg (2002)
- Kleshchev, A., Artemjeva, I.: An analysis of some relations among domain ontologies. Int. Journal on Inf. Theories and Appl. 12, 85–93 (2005)
- Konev, B., Lutz, C., Walther, D., Wolter, F.: Cex and mex: Logical diff and logic-based module extraction in a fragment of owl. Liverpool Uni. and TU Dresden (2008)
- Maedche, A., Staab, S.: Comparing ontologies-similarity measures and a comparison study. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) EKAW 2002. LNCS (LNAI), vol. 2473, p. 251. Springer, Heidelberg (2002)
- Noy, N.F., Musen, M.A.: Promptdiff: A fixed-point algorithm for comparing ontology versions. In: 18th National Conf. on Artificial Intelligence (AAAI) (2002)
- Patel-Schneider, P.F., Hayes, P., Horrocks, I.: Owl web ontology language semantics and abstract syntax. In: W3C Recommendation (2004)
- Qi, G., Hunter, A.: Measuring incoherence in description logic-based ontologies. In: Aberer, K., Choi, K.-S., Noy, N., Allemand, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 381–394. Springer, Heidelberg (2007)
- Volkel, M.: D2.3.3.v2 SemVersion Versioning RDF and Ontologies. EU-IST Network of Excellence (NoE) IST-2004-507482 KWEB (2006)

# ARAGOG Semantic Search Engine: Working, Implementation and Comparison with Keyword-Based Search Engines

Harsh Mittal, Jaspreet Singh, and Jitesh Sachdeva

Netaji Subhas Institute of Technology, Department of Computer Engineering,  
University of Delhi, Delhi, India  
`{Harshmttl,Japji.Singh,Jitesh.Sachdeva}@gmail.com`

**Abstract.** Seeing the exponential growth rate of knowledge and data all over the web, one can foresee that it would be difficult for data driven search engines to cope up with this fast pace of data generation effectively in future. Giving relevant, useful and close to accurate search results will be a big challenge given their current design and approach. Search Engines will be required to make a transit from keyword based search approach to semantic based search approach. It will be equally important for search engines to understand and derive the meaning of user query's actual intent and provide results accordingly. In this backdrop we present our prototype – Aragog, which is even a step ahead than the conventional idea of a semantic search engine. This not only makes the user free from the hassle of browsing through hundreds of irrelevant results, but also generates results in an order that would match its intended context, with a high probability. The main motive behind innovation of Aragog is simple – Focus on user's actual intent of search than number of results. The engine has been designed and tested and the results have been found to be exemplary. In comparison to traditional search engines, it produced more relevant results. Additionally, we have incorporated many other features such as synonym handling and explicit result display that make it all the more tempting to emerge as the next generation's search engine format.

**Keywords:** Semantic Web, Search, Semantic Search, Semantic Page Ranking, Ontology Ranking, Thesaurus.

## 1 Introduction

Since its very inception, the notion of a search engine has been to provide the web users with an interface that could look for appropriate content on the web. The AOL, Google and Yahoo! search engines have all restricted this idea to keyword searching which in the present scenario seems outdated and incomplete. The present generation search engines present a huge amount of search results to the user in response to the query, most of which are at times highly irrelevant, and the user has an ordeal in sifting through these result sets to arrive at some page of his interest.

The limitation of contemporary search engines has forced researchers to look for new alternatives. Semantic engines- which propose to derive meanings out of

sentences seem to fit in place to alleviate out of this hitch. This can be better understood in the light of some examples. For instance, consider a query “*Winner of maiden T20 cricket world cup*”. Intuitively, the user is interested in knowing the direct answer of the query which in this case turns out to be *India*. However, our dry run over some of the conventional search engines yielded us results which cater to the official T20 world cup site, site links for watching T20 world cup, web pages which are flooded with information not worth the user requirement. This clearly highlights the indispensability to look for better alternatives.

A semantic search engine uses ontologies which are a set of concepts mapped together and can be referenced as such to derive semantic associations among different words and concepts. The resources on the web, i.e., the web pages, are crawled and looked for annotations done on them, if any. These annotations are then used to set the words to that ontology with which they have been tagged. These tags are used later on for page searching. The matching and searching algorithms of Aragog have been explained in later sections.

The remaining paper is organized as follows. In Section 2, we discuss about the previous work done in this area. Next we present our motivation towards this work that has been taken up in Section 3. Section 4 talks about our proposed Semantic Search Engine Aragog’s architecture. Section 5 explains the algorithms and the heuristics used in the various modules. Section 6 discusses the results of our implementation of Aragog. Finally we conclude our work and follow that up with the future work that can be done on this engine to enhance its capabilities.

## 2 Previous Works

The architecture for a semantic search engine was proposed in [1]. However, this architecture proposes a two level interaction with the user whereby the user needs to separately mention both the search query and the domain in which the searching shall be performed.

Another important work in this field is [2]. It is a working implementation but it does not incorporate searching according to semantics. It restricts itself to keyword searching in the semantic web documents.

[3] define the Lee’s Model which uses a matching algorithm to reflect the semantic similarity of web page content but it is unable to do the same completely. Thus, it is not possible to satisfy user queries to an appreciable extent.

This all reveals that semantic search engines are still far away from reality and a better solution needs to be found out.

## 3 Motivation

Aragog has been conceived and developed with the following motivations:

1. A Semantic Search Engine should minimize on the number of user interaction levels for better usability. For a particular query, the domain in which the search is to be performed should be deduced automatically. This will bring the Semantic Search Engine at par with the existing keyword based search engine.

2. Synonyms of the keywords should be considered while deriving the semantics of query. Synonyms should be handled in the sense that for a given query, the results of all synonymous queries should also be displayed.
3. Apart from web resources results, a semantic search engine should also provide the user with the exact answer of the query. This would make it a search engine cum answering agent.
4. The query result display should enhance the user experience. This should include ranking amongst the domains and also, ranking within an individual domain. Along with this, relevant text from the web documents should also be displayed to the user.

## 4 Proposed Architecture for Aragog

As discussed in the previous section, the motivation for building Aragog has come from various shortcomings and limitations with other existing semantic search engines. The proposed architecture of Aragog has been shown in Fig. 1.

This architecture supports various features such as ontology ranking synonym handling, semantic answer finder, etc. The various modules are described below:

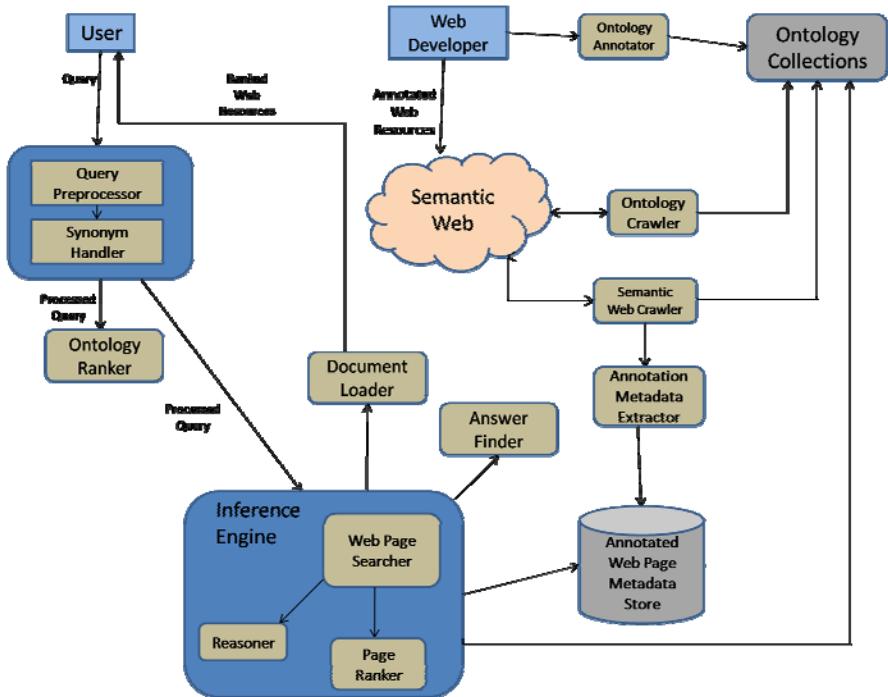
**Query Preprocessor.** This is the module which interacts directly with the user. The query preprocessor is responsible for accepting a query from the user. An acceptance list for this is maintained in a relational database and contains entries for tokens for which corresponding concepts exist in the *ontology collections*. The user's query tokens are verified with the acceptance list and only the tokens which are found in the acceptance list are accepted. Remaining are rejected by the preprocessor. This helps in rejecting helping verbs such as 'is', 'am', 'are' etc. The acceptance list is created/maintained/updated by the Ontology Crawler Module discussed later.

This module also takes care of the scenario when a user query contains similar meaning tokens. For example if the query posed by a user is "*maximum highest score of Sachin Tendulkar in Test*". Here, the concepts '*maximum*' and '*highest*' both correspond to the same meaning. Hence, redundancy is there in the tokens. The query preprocessor also removes similar meaning tokens to avoid complexity and inefficiency that may occur at a later stage. This module also provides features such as support for double quotes in queries.

**Ontology Ranker.** This module is a major improvement over the previously proposed versions. An ontology ranker understands the user's query's context and finds the ontologies which contain the desired concepts. Apart from finding the relevant ontologies, this module also ranks the ontologies for a given query's context.

As explained before, the previous versions expect the user to specify the ontology in which the concepts are to be searched. This module removes this second level interaction with user and hence, increases the responsiveness.

For example, if the user enters the query as "*Tiger Woods*". This query has two contexts. One refers to the golf ontology where *Tiger Woods* is a concept referring to a golf player. The second context refers to forest ontology where *Tiger* and *Woods* are two separate concepts.



**Fig. 1.** Architecture of Aragog semantic search engine

The Ontology Ranker module uses various ontology ranking algorithms and heuristics to rank the ontologies. These have been discussed in later sections.

**Inference Engine/Reasoner.** The Reasoner's task is to infer new information from the given information to help understand the concepts in a better way. This is done by backward chaining to improve on the efficiency front.

For example, if in an ontology for the food Domain, we have a class hierarchy as Food --> Non-Vegetarian Food --> Sea Food. According to the transitive property of reasoning, it can be inferred that *Sea Food* is a type of *Food*.

**Semantic Web Crawler.** The Semantic Web Crawler crawls the web and searches for annotated pages on the web. Annotated pages are the web pages having concepts annotated on them. These annotations are done by the web developer while developing the pages using various tools available. The web crawler then retrieves such pages and passes these pages to the *Annotation Metadata Extractor Module*.

**Ontology Crawler.** This component is responsible for crawling web to find out Ontologies to build Ontology Collection. Apart from finding new Ontologies, this module performs the task of updating existing ontologies on finding new concepts.

**Annotation Metadata Extractor Module.** This module retrieves annotated web pages from the semantic web crawler. It extracts the annotated metadata and concepts from the web page and updates the Metadata Store with the concepts extracted.

**Annotated Web Page Metadata Store.** This store is built by the Web Crawler Module and Annotation Metadata Extractor Module. This store contains the metadata in a relational database and consists of an index for all the annotated concepts in the webpage along with the URLs. The metadata also contains the information to be used by the *Page Ranker* Module.

**Web Page Searcher.** This module is responsible for finding the web resources suitable for the preprocessed query in a particular ontology context.

The searcher refers to the ontology and finds out the relevant concepts to be searched in the metadata store using the various Searcher Algorithms that have been discussed later. These relevant concepts are then searched in the *Annotated Web Page Metadata Store* and the corresponding URLs are retrieved.

**Page Ranker.** This module receives the list of URLs for a given query and ranks these URLs using various page rank algorithms discussed later. The point to be noted here is that the *Page Ranker* ranks only those URLs which correspond to a single ontology domain. Thus, the sorting done here is *Intra-Ontology Sorting* whereas the Ontology Ranker Module does *Inter-Ontology Sorting*.

**Answer Finder.** In certain cases, apart from the query result URLs, it is also helpful to get an answer of the query posed. For example if a user enters a query ‘*Director of Movie Black*’, then along with the related web pages, it is really appreciated if the exact answer i.e. *Sanjay L. Bhansali* is given to the user. Answer Finder Module aims to provide this functionality.

**Semantic Document Loader.** This module is a normal document loader but with extra capability of loading relevant text of a URL depending on the query posed by the user. A local cache copy of each web resource is maintained in the Annotated Web page metadata store that is used by the semantic document loader. For example if a user places the query “*Movies of Shahrukh Khan*”, the conventional search engine’s document loader will display those section of the retrieved pages where either the keyword ‘*Shahrukh Khan*’ or ‘*movies*’ appears, but the Semantic Document Loader will display those sections in the result which contain the name of the movies of Shahrukh Khan.

In the next section, the various algorithms and heuristics adopted for Aragog shall be discussed.

## 5 Proposed Algorithms and Techniques for Various Modules

### 5.1 Ontology Ranker

As stated in previous section, this module ranks the ontologies according to the query based on:

- 1. Presence of Keywords in an Ontology.** Each word present in any of the ontologies has a set of numbers associated with it, where each number corresponds to an ontology. When the user enters a query we compute the intersection of the sets corresponding to each keyword of the query so as to determine the ontology containing all the words of the pre-processed query, which would result in narrowing down the list of ontologies which are required to be searched.

## 2. The Position of Keywords in an Ontology.

- i) If the query consists of a single word, we calculate the depth of the keyword in various ontologies and the one having the keyword at the minimum depth is given the highest priority.
- ii) If the query consists of multiple words, we calculate the Lowest Common Ancestor of the keywords (nodes represented by the keywords). The lowest common ancestor thus found must also be one of the keywords entered. Then the ontology having those keywords separated by minimum distance is ranked first.

### 5.2 Synonym Handler

The synonyms for a word (class name, instance name and property name) are inserted in the same node itself with the help of aliasing. This greatly reduces the time that would have been required in referring to a thesaurus for each word entered in the query.

### 5.3 Semantic Answer Finder

Here the ontology graph is traversed from the least common ancestor of the various nodes to the required instance and the property value of the required property is returned.

### 5.4 Semantic Answer Finder

1. The ontology (graph) is traversed from the property value of the required property (of the required instance) followed by the property name to the least common ancestor of the keywords. A separate set  $S_i$  is constructed for the  $i^{th}$  node of the path, where  $i=0$  corresponds to the property value node.
2. For each set  $S_i$ , a set of links  $L_i$  is searched, such that any of the words of  $S_i$ , appear in the annotation of those web pages.
3. If there were  $n$  nodes in the path, then two sets of sets are computed. Set  $V$  contains  $n$  sets, where the set  $V_i$  is computed from the intersection of all the sets  $L_j$ , where  $i \in [0,n]$  and  $j \in [0,i]$ , and set  $R$  contains  $n-1$  sets, where  $R_k$  is computed from the intersection of sets  $L_m$ , where  $k \in [1,n]$  and  $m \in [1,k]$ .
4. Since, the set  $V$  also incorporates the property values, so the links in the sets of set  $V$  will be preferred over the links in the sets of set  $R$ .
5. Each set  $V_i$  contains the set of links which contains the answer of the query i.e. the property value along with words from the  $1^{st}$  node to the  $i^{th}$  node. Thus,  $V_n$  will give the set of links which are best suited for the query as it would contain the property value and all the keywords of the query (or their synonyms) which were present in the ontology.
6. Therefore, priority will be given to the links of the set  $V_i$  over the links of the set  $V_j$ , where  $i > j$ .
7. Similarly, the links in the set  $R_i$  will be preferred over the links in the set  $R_j$ , where  $i > j$ .

## 6 Implementation of Aragog

A working implementation of Aragog has been developed. In this work we have covered three domains: Bollywood, Cricket and Food. The Ontologies were created for these domains. As OWL-DL is rapidly emerging as a standard to build Ontologies, we have used it to follow the worldwide standards. Protégé tool was used to design Ontologies.

For handling Ontologies and performing operations on them, we used Jena Ontology API. Jena provides us with the Reasoner based on the DL transitive rules. This Reasoner was used in our Inference Engine module.

We chose C#.NET as our development language and ASP.NET was chosen to build the web interface of the Aragog. As Jena API was available in Java, IKVM was used to convert the java source code into a .NET DLL. This development work has been carried on a Intel Dual Core 1.83GHz system having 3GB of DDR2 RAM and 320GB of SATA Hard Disk.

The Aragog search results were compared with Google for a set of queries. The results as presented below clearly highlight how Aragog outperforms the conventional keyword search engine.

### Case 1: Imprecise Queries.

**Query 1.** “Maiden T20 World Cup winner”

**Ideal Result.** All web resources about India preferably in cricket domain.

#### Aragog Results:

**Domain:** Cricket

**Answer:** India

**Top Result:**

<http://www.cricinfo.com/database/NATIONAL/IND/> (Fig. 2)

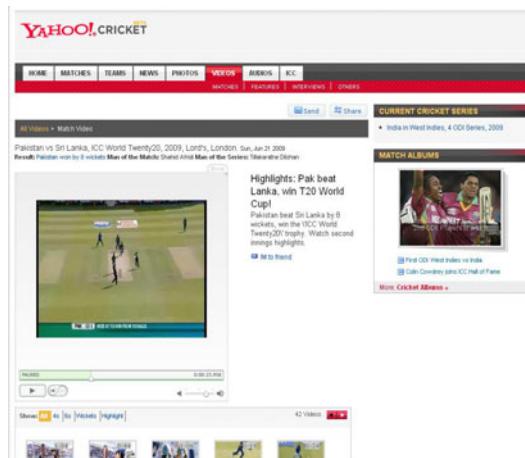
#### Google Results:

**Top Result:**

[http://cricket.yahoo.com/cricket/videos/fvideo/210609\\_SL\\_PAK\\_2inn\\_hl/3222](http://cricket.yahoo.com/cricket/videos/fvideo/210609_SL_PAK_2inn_hl/3222)  
(Fig. 3)



**Fig. 2.** Screenshot of Aragog's top result for query 1



**Fig. 3.** Screenshot of Google's top result for query 1

**Comparison.** It is clearly discernible from the results that Google results returned irrelevant pages which were not desired by the user as shown in Figure 3. On the other hand, Aragog returns a page on a cricket related website about India, as shown in Figure 2, which is first (inferred by ‘maiden’) T-20 world cup winner. Thus, Aragog succeeded in interpreting the correct meaning of the query and displaying the most relevant domain results.

### Case 2: Queries containing some keywords spanning over multiple domains.

**Query 2.** “Ingredients of 20-20”

**Ideal Result.** All pages containing any information regarding the contents of 20-20 biscuits (Brand: Parle-G)

#### Aragog Results:

*Domain:* Food

*Answer:* Wheat, Flour, Sugar, Butter, Milk

*Top Result:*

[http://parleproducts.com/brands/biscuits\\_20-20Cookies.asp](http://parleproducts.com/brands/biscuits_20-20Cookies.asp) (Fig. 4)

#### Google Results:

*Top Result:*

<http://www.mindbodyhealth.com/MbhVision20.htm> (Fig. 5)

**Comparison.** The difference in the quality of results is clearly perceptible. Aragog returns the correct answer page, as shown in Figure 4, whereas Google returns a web page as shown in Figure 5, which is not even remotely related to the food item in question.



**Fig. 4.** Screenshot of Aragog's top result for query 2



**Fig. 5.** Screenshot of Google's top result for query 2

Similarly, several other kinds of queries such as those concerning synonyms, intra domain ranking, inter domain ranking were also tested and the results demonstrate how Aragog outperforms the traditional keyword based search engine.

## 7 Conclusions

The idea of a novel semantic search engine has been proposed as well as implemented in this paper. The results have been found out to be refined, smarter and accurate than the conventional keyword based search engine. Aragog also lays the foundation of semantic search with minimum user intervention. Even the presentation of the results is such that the most apt results are available at a mouse's click. Further, on implementation the proposed Aragog was found to perform much better than the Google search engine.

## 8 Future Work

Aragog leaves us with a few possible future additions that can be made to broaden its searching horizons. Currently, Aragog searches in 3 domains – Bollywood, Cricket and Food. It can easily be extended to incorporate many more domains by adding respective ontologies to Ontology collection.

Aragog can also be easily extended to cover searching amongst videos. This can be done in two ways:

1. Aragog can search through the descriptions of videos available on the internet by simple text search.
2. A speech to text module can also be added to Aragog to allow it to get the contents of the video in a text format which can then easily be searched through to bring semantic search to video contents, which has never been done before.

## References

1. Mudassar Ilyas, Q., Kai, Y.Z., Adeel Talib, M.: A Conceptual Architecture for Semantic Search Engine. In: Proceedings of 8th International Multitopic Conference on INMIC 2004. IEEE Press, Los Alamitos (2004)
2. Ding, L., Finin, T., Joshi, A., Pan, R., Scott Cost, R., Peng, Y., Reddivari, P., Doshi, V., Sachs, J.: Swoogle: A Search and Metadata Engine. In: Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management. ACM Press, New York (2004)
3. Lee, W.P., Tsai, T.C.: An interactive agent-based system for concept-based web search. In: Expert Systems with Applications. Elsevier, Amsterdam (2003)

# Cooperative WordNet Editor for Lexical Semantic Acquisition

Julian Szymański

Gdańsk University of Technology  
Narutowicza 11/12, 80-952 Gdańsk, Poland  
[julian.szymanski@eti.pg.gda.pl](mailto:julian.szymanski@eti.pg.gda.pl)

**Abstract.** The article describes an approach for building WordNet semantic dictionary in a collaborative approach paradigm. The presented system enables functionality for gathering lexical data in a Wikipedia-like style. The core of the system is a user-friendly interface based on component for interactive graph navigation. The component has been used for WordNet semantic network presentation on web page, and it brings functionalities of modification its content by the distributed group of people.

**Keywords:** Lexical semantic, Acquisition, Semantic dictionaries, Collaborative editing, WordNet.

## 1 Introduction

WordNet [8] is one of the largest semantic lexicons of English. It has been developed since 1985 by the Cognitive Science Laboratory at Princeton University. Its authors, based on theories of human cognition, try to reflect all linguistic dependencies between concepts in a common lexical database. The WordNet team has been working on a semantic dictionary for over 22 years. Nowadays<sup>1</sup>, the dictionary contains about 155287 words, organized in 117659 synsets (meaning representations), and includes 206941 pair words – meaning. Introduction of all words with their connections, as well as examples of their usage in language, requires a lot of human work, however the WordNet team has only seven members. The WordNet project has been supported by plenty of grants, which brought together 3 millions dollars. Currently the third release of the WordNet lexical database is available at the project website<sup>2</sup>. WordNet develops as a research project in a closed academic environment. The first version of the dictionary appeared in 1993, and now a third version is available. The dictionary is publicly available, but its modification is restricted to internauts. Probably, the reason for that, is the fact that the lexicon is organized as a set of text files in a specific format, which makes it hard to apply cooperative approach for WordNet development. Lack of cooperative editing functionality is the biggest barrier to scale-up semantic database.

The most well known application of a cooperative approach for gathering data is Wikipedia. The project has experienced great interest from the Internet community

---

<sup>1</sup> ver. 3.0.

<sup>2</sup> <http://wordnet.princeton.edu>

which brought many positive results. Wikipedia has been developed since 2001 by volunteers from all over the world. Currently, the Wikipedia initiative is supported by almost 75000 people, working on over nine million articles written in 125 languages. The largest set of articles is available in English, and contains over 2 million articles.

Nowadays, a lot of projects has been created on the basis of WordNet<sup>3</sup>. They use semantic dictionary as a core knowledge base about language, what enables to implement elementary linguistic competences in a machines.

Some of the implementations do the mapping from WordNet files to other models, especially relational. This can be used to enable a cooperative editing approach.

## 2 Description of WordVenture System

A WordVenture portal<sup>4</sup> has been developed at the Gdansk University of Technology at the Faculty of Electronics, Telecommunications and Informatics. It provides mechanisms for simultaneous work on lexical dictionaries for distributed groups of people and enables cooperative work on aWordNet lexical database. The Cognitive Science Laboratory approach to WordNet development required huge amounts of resources e.g human, time, money [7]. With WordVenture, lexical database development becomes common and cheap.

With WordVenture, a user can browse a WordNet dictionary, and display its content on the screen with a graphical user interface based on an interactive graph. It gives a user-friendly way for visualizing very large sets of contextual data. A user can also query WordVenture to find a specified word and display its senses and related concepts. Connections between nodes (words or senses) are illustrated as edges of a given type. To keep graphs clear, a user can set some constraints to visualize only required types of data. There is also the possibility of interactive graph traversing. Selecting one node all elements that are connected with the marked one are displayed (according to given constraints on data selection).

The advantage brought to WordNet development by WordVenture system is a possibility of editing semantic database by the open, Internet community, which fasten lexical data acquisition process. To provide high quality of the acquired data, all changes introduced by users are represented as change propositions, which are approved or rejected by a privileged user – moderator.

## 3 System Architecture

It was decided that the WordVenture system will be implemented in client-server architecture, with the following assumptions:

- WordNet database and data access logic resides on the server,
- Data visualization mechanisms reside at the client side and provide interfaces to the lexical database in the form of interactive graphs.

---

<sup>3</sup> See: related projects <http://wordnet.princeton.edu/links>

<sup>4</sup> <http://wordventure.eti.pg.gda.pl>

This architecture obliges a developer to implement some functionalities at the server side of the application, but imposes some limitations, especially to communication. The developer has to define communication protocol which will assure flexibility of the data interchange. To widen client-server architecture some elements of the Service Oriented Architecture (SOA) [4] has been introduced. One must meet the following expectations to efficiently implement SOA:

- **Communication Interoperability** – must be assured between different systems and different programming languages. A well-known example of such, is message passing oriented communication [9]. Messages in a defined format are sent between sender and receiver, who performs content-based computations. Neither receiver nor sender have to have precise knowledge about the other's side of a platform.
- **Publishing, Discovery and Service Inquiry** – these are basic concepts of SOA architecture. The three operating sides can be distinguished: **service provider** (creates and publishes his services – service producer), **service broker** (gives mechanisms to store information about services e.g. physical location of remote service and performs search operations), and **service requester** (invokes remote service – service consumer).

One of the most popular implementations of SOA are web services. They've met above requirements, especially communication interoperability between many development platforms e.g. J2EE and Microsoft .NET. Every web service is described in a well-defined and common language – WSDL (Web Service Description Language) [3] and it uses SOAP (Simple Object Access Protocol) as a transport protocol [10]. In SOAP, messages are passed as XML documents.

The original implementation of a WordNet database uses text files. Because of their structure, modification is available only with dedicated tools. This type of storage doesn't support synchronous access for modification, nor allows to perform efficiently large amount of queries. It also requires us to create special mechanisms for editing, including synchronization and file structure refactoring, after any operation. To enable editing of a WordNet lexical database we had to perform mappings between WordNet text files and a relational database. Transformation from text files to its relational representation was performed by the WordNet SQL Builder tool<sup>5</sup>.

### 3.1 Server-Side Architecture

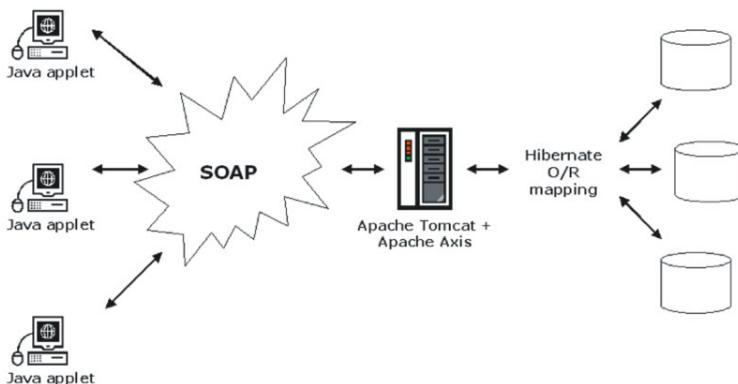
The server-side of the WordVenture application makes its functionalities available through web services. According to communication interoperability requirement, it is possible to connect client application that can be implemented in different technologies. Web services have been developed and deployed with the Apache Axis framework<sup>6</sup>, which resides in the servlets container – Apache Tomcat<sup>7</sup>. Apache Axis framework is a set of libraries and tools which allows a developer to create and publish web services. Axis is just an ordinary web application that can be deployed to any servlet container,

---

<sup>5</sup> <http://wnsqlbuilder.sourceforge.net>

<sup>6</sup> <http://ws.apache.org/axis>

<sup>7</sup> <http://tomcat.apache.org>



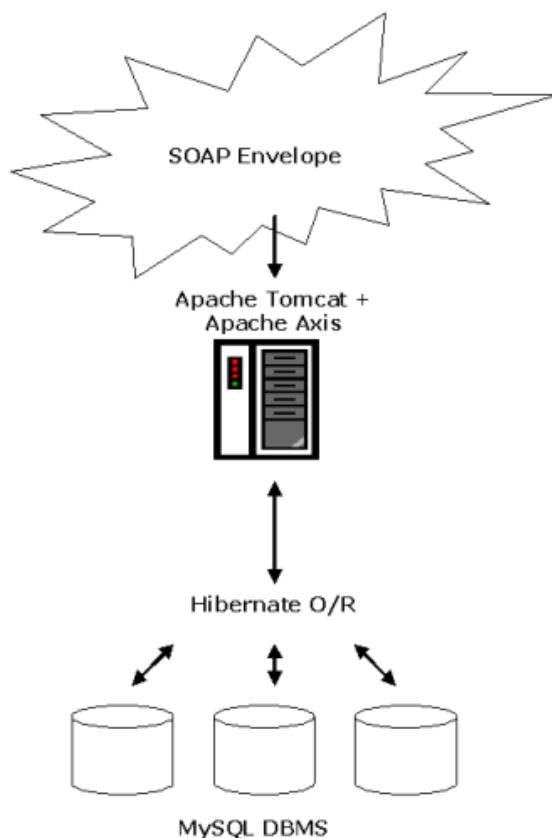
**Fig. 1.** Basic concept of the WordVenture architecture and its elements

especially to Apache Tomcat. It listens for a request from client application that is sent as a SOAP envelope. When a message comes, Axis interprets it and calls a local procedure. Subsequently, a response message is created and sent to the client application. Apache Axis allows programmers to deploy web services such as Plain Old Java Objects (POJOs), which have to have changed extension (from .java to .jws). Deployment can be done by copying jws (Java Web Service) file to proper Axis directory.

The second edition of WordVenture [11] introduce mechanisms which allows an moderator to control user actions. To control modifications of lexical database authentication and authorization mechanisms have been created. An anonymous user can only browse the data from WordNet database, if he wants to edit it, he must log in. Every modification introduced to database is represented as „change proposition” and is sent to the moderator. This moderator, as a privileged user, can commit or reject every modification proposed by an ordinary user.

Every server functionality allows a user to perform three different groups of actions depending on the role that user has:

- **Functionalities for Browsing WordNet Lexical Database** – are available to every user (anonymous and logged-in). After invoking an action on the client-side of application, a proper remote procedure is called on server. The server queries database and sends data to client application. Because of efficiency reasons, all data that is sent between sender and receiver is serialized and compressed, so transmission through the Internet is much more faster.
- **Functionalities that Allows a User to Edit WordNet Lexical Database** – are only available to registered users. After invoking an edit action on the client-side of an application, the proper change proposition is created. Subsequently, this proposition is sent to the server to be added to database. A privileged user (moderator) can view all change propositions and select commit, other cancel. After committing, a proposition is permanently added to database and can be seen by other users.
- **Administrative Functionalities which are Connected with User Management** – are available only for privileged users – administrators. They can perform user deletion or user rights editing in WordVenture system. Every administrator can give administrative rights to another user.

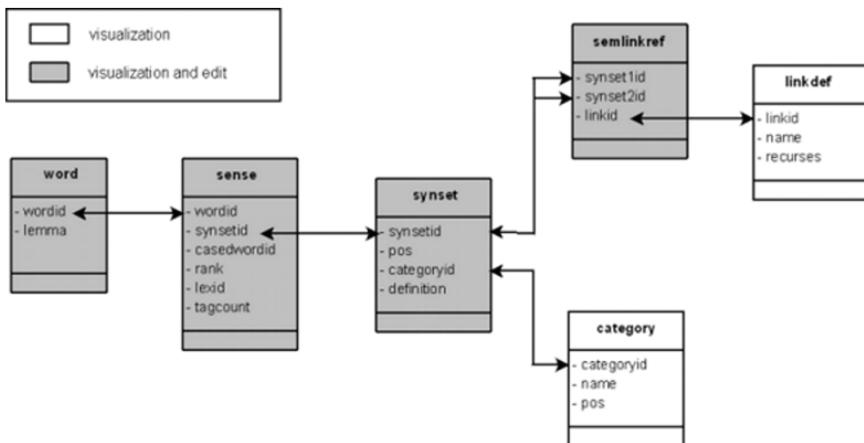


**Fig. 2.** Main view of server-side architecture

It was decided to use an object-relational mapping mechanism to make our system reusable. Almost all of the object-relational mapping engines use DAO objects (Data Access Object). This project pattern allows the developer to separate data access logic from logic operating on those data. Object-relational mapping mechanisms enable developer to translate data, from relational structure to object structure, what keep proper relations. Each row in a table is translated to a proper object. We've use Hibernate as O/R mapping engine<sup>8</sup> what is a highly-developed and effective solution. One of its main features is the "lazy loading" mechanism [1]. It prevents from retrieving at one time all data from database (which can be very inefficient). Lazily loaded mechanisms get data from database only when a end-user wants to see it.

The Figure 2 presents a detailed diagram of server side architecture, which includes all the above-mentioned technological solutions. It shows how the server handles SOAP messages sent by client application. Web service invocation starts when a SOAP envelope comes to the server. Apache Axis framework, resides on Apache Tomcat servlet container, and is responsible for handling SOAP messages. Creating a new web service

<sup>8</sup> <http://www.hibernate.org>



**Fig. 3.** WordNet entities supported by the tool. Grayed out entities have support for both visualization and editing, white entities have only visualization support. Arrows represent relationships between entities.

in Java, from the developers' point of view, requires programming public class with public methods and deploying it to Apache Axis. In a WordVenture system those public classes are used to exploit the Hibernate O/R mapping engine to access database and perform all required queries.

The elements of the WordNet like a word position or morphological definitions are not as much necessary as lemmas and synsets. To simplify the editing process, it was decided to allow only for modification of the semantic net structure. The database structure for handling data provided by WordVenture is presented in the Figure 3, where editable and dictionary tables of the system are shown.

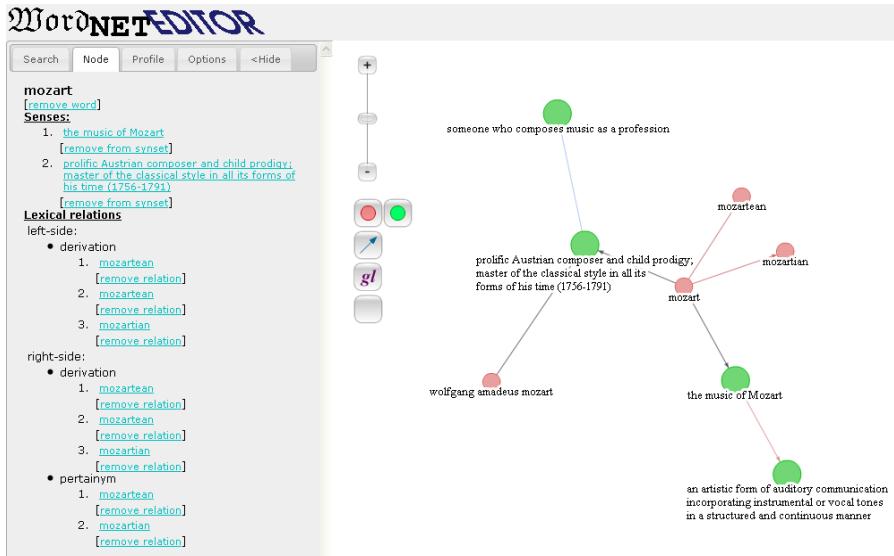
### 3.2 Client-Side Architecture

WordVenture has been developed in rich-client architecture [2]. Because of that, some logic connected with data visualization, can be executed on the client-side of application. The first implementation of the system has been based on modified TouchGraph component<sup>9</sup> for interactive graph visualization. In present version of the system we use our own component called Gossamer<sup>10</sup> which allow to visualize large graphs using Flash technology. The system allows a user to:

- **Browse WordNet Lexical Database** – using interactive interface (see Figure 4) the user is able to navigate over the WordNet semantic network in a user-friendly way. Words and synsets are visualized as graph nodes, connections between them are visualized as graph edges. Additionally, the user can filter graph nodes and edges to obtain required content (according to a selected type), what makes user interface clean and readable.

<sup>9</sup> <http://www.touchgraph.com>

<sup>10</sup> <http://gossamer.eti.pg.gda.pl>



**Fig. 4.** Sample visualization of the WordNet semantic network

- **Perform Modifications on WordNet Lexical Database** – the tool enables a user to change graph content by adding, editing, or deleting its elements: nodes and edges. Modification of above-mentioned elements of WordNet lexicon (see Figure 3) does not cover all components of WordNet. It only covers the four most desired, from the user point of view, elements of the semantic network: words, synsets, senses and relations, presented in the Figure 3.

Modification of WordNet lexicon is based on well-known rules from other cooperative projects like Wikipedia [14]:

- **Changes Patrolling** – every modification of WordNet lexicon is represented as a change proposition that is sent to a privileged user – moderator, who can commit or reject the proposition. This approach is used to trace every activity performed by the cooperative community. Such a mechanism can be used to detect undesirable users’ activities: vandalism, violation of copyrights and others.
- **“Free” Character of Wikipedia** – every interested user can join the WordVenture community and cooperate with its creation.

In the WordVenture system a user is able to use the context menu which is available under right click of mouse. Selecting a word or synset makes the system show options available to choose. Functionality of WordNet lexicon editing in cooperative paradigm [15] is available only for a logged-in user. In previous release of the system [11] synchronous work of many users caused saving only of the last modification. From now on, every modification is saved as a change proposition, and is sent to an moderator. He can choose whether a proposition is permanently saved, or deleted.

Graph-based visualization in a WordVenture system allows a user to work efficiently, and keep clean and readable a large amount of lexical data. In every moment a user can

enable or disable required elements of the visualization, which makes his workspace personalized. Additionally, it is possible to zoom in or zoom out view of graph, so a user is able to keep a lot of graph nodes on his workspace.

## 4 Cooperative Approach to Building Lexical Nets

Lack of tools for cooperative editing of semantic dictionary databases is the main barrier for rapid WordNet development. Our mission is to deliver a tool enabling a cooperative editing approach for many users placed in distributed Internet environment. Cooperative editing is connected with publishing the WordNet database and making it open to the Internet community. This brings advantages for faster WordNet development, however some problems may arise:

- **Vandalism** – may cause loss of all important data, kept in current release of lexical database. It also can affect the data structure e.g. creating pointless connections between words and senses. Because of that, it is important to deliver tools which will reduce the risk of the above-mentioned.
- **Simultaneous** work on the same part of database, by many users, may reveal some conflicts resulting from concurrent work of many users at the same time. In the worst case, one user can add connection to an element of the WordNet dictionary that was deleted by another.

The best solution of these above-mentioned problems is to introduce the role of privileged user – moderator. He or she is able to see every change that is proposed to the lexical database dictionary. Every user, after logging in, can edit the lexical database in a restricted way. All introduced modifications are represented as change propositions that are sent to an moderator, who can browse them, and decide whether propositions can be added to database or deleted. All administrative actions result in permanent semantic network update. This approach will also allow us to save history of the database modifications and to detect users – vandals, whose rights can be permanently taken back. According to the basic rule of effective team work („communication, coordination and cooperation”) [5], the users were delivered the possibility of continuous communication via a web-based forum. While using it, users can define their own strategies for WordNet lexical database development, reach their own conclusions and also feel all advantages of synergic effect.

The current release of WordVenture system includes all above-mentioned functionalities and is available on project web site: <http://wordventure.eti.pg.gda.pl>. At present, we are evaluating future proposals for the system, gathering more feedback from users via our web-based forum system, prioritizing future goals, and evaluating the applied solution as a base for a generic approach to semantic data editing tasks.

## 5 Conclusions

Our project has been developed and successfully deployed. Currently the WordVenture system has been extended to introduce mechanisms avoiding problems connected with cooperative editing approach:

- Authentication, authorization, and logging users activity – while switching to editing mode, a user is asked to fill-in an authentication form. After logging-in, a user is able to create change propositions that are sent to an moderator. He can trace all the changes and decide whether to approve or reject them.
- Tracing users' change proposals – a privileged user has rights to manage change propositions introduced by other users. Because of that, it is possible to avoid all unwanted changes, and also to review all proposals by a qualified person.

The WordVenture system starts from the newest version of WordNet lexical database (3.0). The system architecture allows us to perform trouble-free actualization of dictionary version with assumption that data structure will not change.

Offering the cooperative editing of the dictionary for the Internet community, seems to be a very attractive way for gathering lexical semantics. It creates the opportunity for fast semantic dictionary development with the cooperation of people from all over the world. It takes down all the duties put on the team while creating the next versions of WordNet dictionary as well. However, we should remember about potential threats which can arise while opening the dictionary for the wide Internet community. The system have been developed based on the experience of Wikipedia. In the current version of the system the risk of vandalism or the unintentional destruction of content has been eliminated, which makes a cooperative approach more reliable.

## 6 Future Mission

The WordVenture system has reached the end of its second iteration. In this section we propose changes that will be applied in next versions of WordVenture. Next iteration can include improvements as follows:

1. Internationalization of client-side application – all inscriptions should be organized as supply and should be translated.
2. Integration other lexical networks to WordVenture to make linguistic database richer. By now we consider two projects: Microsoft MindNet [13], and ConceptNet [6].
3. Extension of user activities tracking and edit functionalities to other elements of WordNet database.
4. Introducing improvements to the user interface, especially to administrative part. Currently, an moderator has to manually merge some changes entered by user.
5. Extending the search engine: search by keywords in synset descriptions, etc.
6. Integration of Wikipedia and WordVenture semantic network [12] for details see <http://swn.eti.pg.gda.pl>.

Future development of WordVenture depends also on users' opinions received via our web based forum. We are waiting for any suggestions and comments about WordVenture – development ideas are welcome. We want to invite everyone to use our system and give us feedback.

**Acknowledgements.** This work was supported by Polish Ministry of Science and Higher Education under research project N N516 432 338.

## References

1. Arendt, J., Giangarra, P., Manikundalam, R., Padgett, D., Phelan, J.: System and method for lazy loading of shared libraries, uS Patent 5,708,811 (1998)
2. Boudreau, T., Tulach, J., Wielenga, G.: Rich client programming: plugging into the netbeans platform (2007)
3. Christensen, E., Curbera, F., Meredith, G., Weerawarana, S.: Web services description language (WSDL). W3C Web Site (2001)
4. Erl, T.: Service-oriented architecture: concepts, technology, and design. Prentice Hall PTR, Upper Saddle River (2005)
5. Kling, R.: Cooperation, coordination and control in computer-supported work. Communications of the ACM 34(12), 83–88 (1991)
6. Liu, H., Singh, P.: ConceptNet?a practical commonsense reasoning tool-kit. BT Technology Journal 22(4), 211–226 (2004)
7. Miller, G., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: Introduction to wordnet: An on-line lexical database\*. International Journal of lexicography 3(4), 235–244 (1990)
8. Miller, G., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: WordNet: An on-line lexical database. International Journal of Lexicography 3(4), 235–312 (1990)
9. Palmer, R., Gopalakrishnan, G., Kirby, R.: The communication semantics of the message passing interface. Technical Report UUCS-06-012, The University of Utah (2006)
10. Scribner, K., Scribner, K., Stiver, M.: Understanding Soap: Simple Object Access Protocol. Sams, Indianapolis (2000)
11. Szymański, J., Dusza, K., Byczkowski, Ł.: Cooperative Editing Approach for Building Wordnet Database. In: Proceedings of the XVI International Conference on System Science, pp. 448–457 (2007)
12. Szymbaski, J., Kilanowski, D.: Wikipedia and wordnet integration based on words co-occurrences. In: Proceedings of International Conference on System Science and Technology (2009)
13. Vanderwende, L., Kacmarcik, G., Suzuki, H., Menezes, A.: MindNet: an automatically-created lexical resource. HLT/EMNLP. The Association for Computational Linguistics (2005)
14. Viegas, F., Wattenberg, M., Kriss, J., Van Ham, F.: Talk before you type: Coordination in Wikipedia. In: Hawaii International Conference on System Sciences, vol. 40, p. 1298. IEEE, Los Alamitos (2007)
15. Yang, Y., Sun, C., Zhang, Y., Jia, X.: Real time cooperative editing on the Internet. IEEE Internet Computing 4(3), 18–25 (2000)

# Analogical Cliques in Ontology Construction

Guofu Li and Tony Veale

School of Computer Science and Informatics  
University College Dublin, Ireland  
[{guofu.li,tony.veale}@ucd.ie](mailto:{guofu.li,tony.veale}@ucd.ie)  
<http://afflatus.ucd.ie>

**Abstract.** If we view ontology-matching and analogical-mapping as different perspectives on the same structural processes, then it follows that matching can sensibly be applied both *between* ontologies, to ensure inter-operability, and *within* ontologies, to increase internal symmetry. When applied within a single ontology, matching should allow us to identify pockets of structure that possess higher-order similarity that is not explicitly rejected in the ontology's existing category structure. This paper explores how cliques of analogies (or *analogical cliques*) can be used to support the creation of a new layer of structure in an ontology, to better reject human intuitions about the pragmatic similarity of different categories and entities.

**Keywords:** Analogy, Mapping, Cliques, Ontology induction.

## 1 Introduction

Ontologies, like languages, are meant to be shared. A common ontology allows multiple agents to share the same specification of a conceptualization [1], ensuring mutual intelligibility when communicating in the same domain of discourse. But like languages, there are often many to choose from: each ontology is a man-made artifact that reflects the goals and perspective of its engineers [2], and different ontologies can model a domain with differing emphases, at differing levels of conceptual granularity. Inevitably, then, multiple agents may use different ontologies for the same domain, necessitating a mapping between ontologies that permits communication, much like a translator is required between speakers of different languages.

Given the operability problems caused by semantic heterogeneity, the problem of matching different ontologies has received considerable attention in the ontology community (*e.g.* [3]). Fortunately, formal ontologies have several properties that make matching possible. Though formal in nature, ontologies can also be seen as ossified linguistic structures that borrow their semantic labels from natural language [4]. It is thus reasonable to expect that corresponding labels in different ontologies will often exhibit lexical similarities that can be exploited to generate match hypotheses. Furthermore, since ontologies are highly organized structures, we can expect different correspondences to be systematically related. As such, systems of matches that create isomorphisms between the local structures of different ontologies are to be favored over bags of unrelated matches that may lack coherence. In this respect, ontology matching

has much in common with the problem of analogical mapping, in which two different conceptualizations are structurally aligned to generate an insightful analogy [5]. Indeed, research in analogy [5] reveals how analogy is used to structurally enrich our knowledge of a poorly-understood domain, by imposing upon it the organization of one that is better understood and more richly structured. Likewise, the matching and subsequent integration of two ontologies for the same domain may yield a richer model than either ontology alone.

This paper has several related goals. First, we demonstrate how analogical mappings can be derived from corpora for large ontologies that are themselves induced via text analysis. Second, we show how this system of analogical mappings can itself be subjected to further structural analysis, to yield *cliques* of related mappings. Third, we show how cliques can act as higher-level categories in an ontology, to better capture the intuitions of end-users (as reflected in their use of language) about which categories and entities are more similar than others.

We begin in section 2 with a consideration of the clustering role of categories in ontologies, and how the graph-theoretic notion of a clique can also fulfil this role, both at the level of instances and categories. In section 3 we describe the induction of our test ontology, called *NameDropper*, from the text content of the web. In section 4 we then show how analogical mappings between the categories of *NameDropper* can also be extracted automatically from web content. This network of analogical mappings provides the grist for our clique analysis in section 5, in which we show how *ontological cliques* — tightly connected groupings of analogical mappings between ontological categories — can be created to serve as new upper-level category structures in their own right. We conclude with some final thoughts in section 6.

## 2 Categories and Cliques

The taxonomic backbone of an ontology is a hierarchical organization of categories that serves to cluster ideas (both sub-categories and instances) according to some intrinsic measure of similarity. In the ideal case, ideas that are very similar will thus be closer together — *i.e.*, clustered under a more specific category — than ideas that have little in common. Ontologies give this hierarchy of categories an explicit logical structure, wherein categories are defined according to the shared properties that make their members similar to each other. Ontologies that employ more categories can thus make finer distinctions that better reflect the semantic intuitions of an end-user [6].

Compare, for instance, the taxonomy of noun-senses used by WordNet [7] with that of HowNet [8]. In WordNet, the category of {human, person} is divided into a few tens of sub-types, which are themselves further sub-divided, to hierarchically organize the different kinds and roles of people that one might encounter. In HowNet, however, every possible kind of person is immediately organized under the category *Human*, so that thousands of person-kinds share the same immediate hypernym. For this reason, WordNet offers a more viable taxonomic basis for estimating the semantic similarity of two terms [9].

Nonetheless, it is important to distinguish between semantic similarity and pragmatic comparability. The measures described in [9] estimate the former, and assign a

similarity score to any pair of terms they are given, no matter how unlikely it is that a human might every seek to compare them. Comparability is a stronger notion than similarity: it requires that a human would consider two ideas to be drawn from the same level of specificity, and to possess enough similarities and differences to be usefully compared. There is thus a pragmatic dimension to comparability that is difficult to express in purely structural terms. However, we can sidestep these difficulties by instead looking to how humans use language to form clusters of comparable ideas. This will allow us to replace the inflexible view of ontological categories as clusters of semantically-similar ideas with the considerably more flexible view of categories as clusters of pragmatically-comparable ideas.

It has been widely observed that list-building patterns in language yield insights into the ontological intuitions of humans (e.g., [6], [10], [11]). For instance, the list “*hackers, terrorists and thieves*”, which conforms to the pattern “*Nouns, Nouns and Nouns*”, tells us that *hackers*, *terrorists* and *thieves* are all similar, are all comparable, and most likely form their own sub-category of being (e.g., such as a sub-category of *Criminal*). We can build on this linguistic intuition by collecting all matches for the pattern “*Nouns and Nouns*” from a very large corpus, such as the Google n-grams [12], and use these matches to create an adjacency matrix of comparable terms. If we then find the maximal cliques that occur in the corresponding graph, we will have arrived at a pragmatic understanding of how the terms in our ontology should cluster into categories if these categories are to reflect human intuitions.

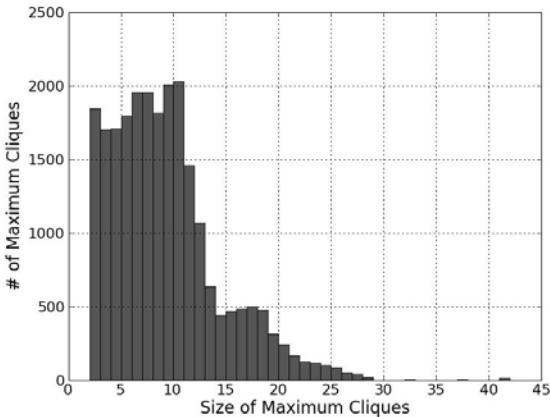
A clique is a complete sub-graph of a larger graph, in which every vertex is connected to every other [13]. A  $k$ -clique is thus a fully-connected sub-graph with  $k$  vertices, while a clique is *maximal* if it is not a proper-subset of another clique in the same graph. In ontological terms then, a clique can represent a category in which every member has an attested affinity with every other, *i.e.*, a category in which every member can be meaningfully compared with every other. Since all ontologies are graphs, cliques have a natural semantic resonance in ontologies, leading some authors [14] to propose cliques as a graph-theoretic basis for estimating the similarity of two ontologies.

Cliques also indicate similarity within ontologies. Figure 1 shows the distribution of maximal clique sizes that we find when using the “*Nouns and Nouns*” pattern in the Google n-grams to mine coordinated pairs of capitalized terms. In general, the cliques correspond to proper subsets of existing categories, and mark out subsets whose members are more similar to each other than to other members of the larger category. For instance, we find this 11-clique:

$$\{ \textit{Environment, Education, Finance, Industry, Health, Agriculture, Energy, Justice, Science, Defence, Transport} \}$$

This clique seems to cluster the key societal themes around which governments typically structure themselves, thus suggesting an ontological category such as *Government Ministerial Portfolio*.

Since the notion of a clique is founded on a social metaphor, an example concerning proper-named entities can be illustrative. Using the Google n-grams and a named-entity



**Fig. 1.** Cliques of different sizes in the graph of coordinated nouns found in the Google n-grams corpus

detector, we can build an adjacency matrix of co-occurring entities and derive from the resulting graph a set of maximal cliques. One such clique is the following 4-clique:

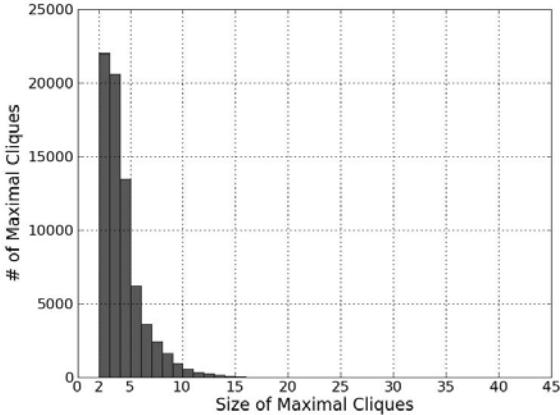
$$\{Steve\ Jobs, Bill\ Gates, Michael\ Dell, Larry\ Ellison\}$$

In an ontology of proper-named entities, such as the *NameDropper* ontology described in the next section, we would expect these entities to all belong to the category *CEO*. However, this category is likely to have thousands of members, so many additional sub-categories are needed to meaningfully organize this space of *CEOs*. What makes these particular *CEOs* interesting is that each is an iconic founder of a popular technology company; thus, they are more similar to each other than to *CEOs* of other companies of comparable size, such as those of *GE*, *Wal-Mart* or *Pfizer*. In the ideal ontology, these entities would be co-members of the more specific category *TechCompany-CEO*.

As shown in Figure 2, large cliques (*e.g.*,  $k > 10$ ) are less common in the graph of co-occurring propernamed instances than they are in the graph of co-occurring categories (Figure 1), while small cliques are far more numerous, perhaps detrimentally so. Consequently, we find many partially overlapping cliques that should ideally belong to the same fine grained category, such as *Irish-Author*:

$$\begin{aligned} &\{Samuel\ Beckett, James\ Joyce, Oscar\ Wilde, Jonathan\ Swift\} \\ &\{Samuel\ Beckett, Bram\ Stoker, Oscar\ Wilde, Jonathan\ Swift\} \\ &\quad \{Samuel\ Beckett, Seamus\ Heaney\} \\ &\quad \{Patrick\ Kavanagh, Brendan\ Behan, James\ Joyce\} \end{aligned}$$

This fragmentation presents us with two possible courses of action. We can merge overlapping cliques to obtain fewer, but larger, cliques that are more likely to correspond to distinct sub-categories. Or we can apply clique analysis not at the level of instances, but at the level of categories themselves. In this paper we shall explore the latter option.



**Fig. 2.** Cliques of different sizes from the graph of coordinated proper-names in the Google n-grams corpus

### 3 NameDropper Ontology

As a test-bed for our explorations, we choose a domain in which the notion of a clique has both literal and metaphoric meaning. *NameDropper* is an ontology of the proper-named concepts — such as people, places, organizations and events — that one would expect to find highlighted in an online newspaper. *NameDropper* is used to semantically annotate instances of these entity-kinds in news-texts and to provide a series of one or more (preferably many) categorizations for each instance.

Categories in *NameDropper* are semantically-rich, and serve as compressed propositions about the instances they serve to organize. For instance, rather than categorize *Steve Jobs* as a *CEO*, we prefer to categorize him as *Apple CEO* or *Pixar CEO*; rather than categorize *Linus Torvalds* as a developer, we categorize him as a *Linux developer* and a *Linux inventor*; and so on. In effect then, each category is more than a simple generalization, but also encodes a salient relationship between its instances and other entities in the ontology (*e.g.*, *Linux*, *Apple*, *etc.*). As we show in the next section, this use of a rich-naming scheme for categories means that analogies between different categories can be identified using simple linguistic analysis of category labels. It also happens that these kinds of rich names are more readily and reliably extracted from short text n-grams than more overt propositions (*e.g.*, such as “*Steve Jobs is the CEO of Apple*”).

The *NameDropper* ontology is extracted from the text of the Google n-grams in a straightforward manner. Simply, we focus on apposition patterns of the following form:

1. *Mod Role Firstname Lastname*
2. *Mod<sub>1</sub> Mod<sub>2</sub> Role Firstname Lastname*
3. *Mod Role Firstname Midname Lastname*
4. *Mod<sub>1</sub> Mod<sub>2</sub> Role Firstname Midname Lastname*

Here *Mod*, *Mod<sub>1</sub>* or *Mod<sub>2</sub>* is any adjective, noun or proper-name, *Firstname*, *Midname*, and *Lastname* are the appropriate elements of a named entity, and *Role* is any

noun that can denote a position, occupation or role for a named-entity. A map of allowable name elements is mined from WordNet and Wikipedia, while a large list of allowable *Role* nouns is extracted from WordNet by collecting all single-term nouns categorized as *Workers*, *Professionals*, *Performers*, *Creators* and *Experts*. Since pattern (4) above can only be extracted from 6-grams, and Google provides 5-grams at most, we use overlapping 5-grams as a basis for this pattern.

When applied to the Google n-grams corpus, these patterns yield category/instance pairs such as:

- a. “*Gladiator director Ridley Scott*”
- b. “*Marvel Comics creator Stan Lee*”
- c. “*JFK assassin Lee Harvey Oswald*”
- d. “*Science Fiction author Philip K. Dick*”

Of course, not all pattern matches are viable category/instance pairs. Importantly, the patterns *Mod Role* or *Mod<sub>1</sub> Mod<sub>2</sub> Role* must actually describe a valid category, so partial matches must be carefully avoided. For instance, the following matches are all rejected as incomplete:

- \*e. “*Microsystems CEO Scott McNealy*”
- \*f. “*Vinci code author Dan Brown*”
- \*g. “*Meeting judge Ruth Bader Ginsberg*”
- \*h. “*The City star Sarah Jessica Parker*”

The n-grams in examples \*e, \*f and \*h are clearly truncated on the left, causing a necessary part of a complex modifier to be omitted. In general this is a vexing problem in working with isolated n-grams: it is difficult to know if the n-gram stands alone as a complete phrase, or if some key elements are missing. In example \*g we see that *Meeting* is not a modifier for *judge*, but a verb that governs the whole phrase. Nonetheless, we deal with these problems by performing the extraction and validation of category labels prior to the extraction of category/instance appositions. The following patterns are thus used to extract a set of candidate category labels from the Google n-grams:

5. *the Mod Role*
6. *the Mod<sub>1</sub> Mod<sub>2</sub> Role*
7. *the Role of Mod<sub>1</sub> Mod<sub>2</sub> (→ Mod<sub>1</sub> Mod<sub>2</sub> Role)*
8. *the Role of Mod (→ Mod Role)*

The patterns allow us to identify the collocations “*the CEO of Sun Microsystems*” (via 7) and “*the Supreme Court judge*” (via 6) as valid categories but not “*the Meeting judge*” or “*the Microsystems CEO*” (which are not attested). Thus, only those collocations that can be attested via patterns 5 ~ 8 in the Google n-grams are allowable as categories in the patterns 1 ~ 4.

Overall, the intersection of patterns 1 ~ 4 and 5 ~ 8 extracts over 200,000 different category/instance pairings from the Google n-grams corpus, ascribing an average of 6 categories each to over 29,000 different named-entity instances. Because the Google corpus contains only those n-grams that occur 40 times or more on the web, the extraction process yields remarkably little noise. A random sampling of *NameDropper*'s contents suggests that less than 1% of categorizations are malformed.

## 4 Analogical Mappings

These patterns lead *NameDropper* to be populated with many different complex categories and their proper-named instances; each complex category, like *Apollo 11 astronaut*, is a variation on a basic role (*e.g.*, *astronaut*) that serves to link an instance (*e.g.*, *Neil Armstrong*) to this role in a specific context (*e.g.*, *Apollo 11*). There is some structure to be had from these complex categories, since clearly, an *Apollo 11 astronaut* is an *Apollo astronaut*, which in turn is an *astronaut*. But such structure is limited, and as a result, *NameDropper* is populated with a very broad forest of shallow and disconnected mini-taxonomies. The ontology clearly needs an upper-model that can tie these separate category silos together, into a coherent whole. One can imagine WordNet acting in this capacity, since the root term of every mini-taxonomy is drawn from WordNet's noun taxonomy. Yet, while WordNet provides connectivity between basic roles, it cannot provide connectivity between complex categories.

For instance, we expect *Apollo astronaut* and *Mercury astronaut* to be connected by the observation that *Apollo* and *Mercury* are different *NASA programs* (and different *Greek Gods*). As such, *Apollo astronaut* and *Mercury astronaut* are similar in a different way than *Apollo astronaut* and *American astronaut*, and we want our ontology to reflect this fact. Likewise, *Dracula author* (the category of *Bram Stoker*) and *Frankenstein author* (the category of *Mary Shelley*) are similar not just because both denote a kind of author, but because *Dracula* and *Frankenstein* are themselves similar. In other words, the connections we seek between complex categories are analogical in nature. Rather than posit an ad-hoc category to cluster together *Dracula author* and *Frankenstein author*, such as *Gothic monster novel author* (see [15] for a discussion of ad-hoc categories), we can use an analogical mapping between them to form a cluster.

But as can be seen in these examples, analogy is a knowledge-hungry process. To detect an analogy between *Apollo astronaut* and *Mercury astronaut*, a system must know that *Apollo* and *Mercury* are similar programmes, or similar gods. Likewise, a system must know that *Dracula* and *Frankenstein* are similar books to map *Dracula author* to *Frankenstein author*. Rather than rely on WordNet or a comparably large resource for this knowledge, we describe here a lightweight corpus-based means of finding analogies between complex categories.

Two complex categories may yield an analogy if they elaborate the same basic role and *iff* their contrasting modifier elements can be seen to belong to the same semantic field. The patterns below give a schematic view of the category mapping rules:

1.  $Mod_X \text{ Role} \rightarrow Mod_Y \text{ Role}$
2.  $Mod_X \text{ Mod Role} \rightarrow Mod_Y \text{ Mod Role}$
3.  $Mod \text{ Mod}_X \text{ Role} \rightarrow Mod \text{ Mod}_Y \text{ Role}$
4.  $Mod_A \text{ Mod}_B \text{ Role} \rightarrow Mod_X \text{ Mod}_Y \text{ Role}$

*E.g.*, these rules can be instantiated as follows:

1. *Java creator* → *Perl creator*
2. *Apple inc. CEO* → *Disney inc. CEO*
3. *Apollo 11 astronaut* → *Apollo 13 astronaut*
4. *Man United striker* → *Real Madrid striker*

Clearly, the key problem here lies in determining which modifier elements occupy the same semantic field, making them interchangeable in an analogy. We cannot rely on an external resource to indicate that *Java* and *Perl* are both languages, or that *Apple* and *Disney* are both companies. Indeed, even if such knowledge was available, it would not indicate whether a human would intuitively find *Java* an acceptable mapping for *Linux*, say, or *Apple* an acceptable mapping for *Hollywood*, say. What is an acceptable level of semantic similarity between terms before one can be replaced with another?

Fortunately, there is a simple means of acquiring these insights automatically. As noted in section 2, coordination patterns of the form *Noun*<sub>1</sub> and *Noun*<sub>2</sub> reflect human intuitions about terms that are sufficiently similar to be clustered together in a list. For instance, the following is a subset of the Google 3-grams that match the pattern "Java and \*":

"Java and Bali", "Java and C++", "Java and Eiffel"  
 "Java and Flash", "Java and Linux", "Java and Perl"  
 "Java and Python", "Java and SQL", "Java and Sun"

Coordination typically provides a large pool of mapping candidates for a given term. To minimize noise, which is significant for such a simple pattern, we look only for the coordination of capitalized terms (as above) or plural terms (such as "cats" and "dogs"). Much noise remains, but this does not prove to be a problem since substitution of comparable terms is always performed in the context of specific categories. Thus, *Perl* is a valid replacement for *Java* in the category *Java creator* not just because *Java* and *Perl* are coordinated terms in the Google 3-grams, but because the resulting category, *Perl creator*, is a known category in *NameDropper*. As a result, *James Gosling* (*Java creator*) and *Larry Wall* (*Perl creator*) are analogically linked. Likewise, *Linux creator* and *Eiffel creator* are valid analogies for *Java creator*, but not *Bali creator* or *Sun creator*, since these are not known categories.

Since categories can have multi-word modifiers (e.g., *King Kong director*, *Harry Potter star*), we run a range of patterns on Google 3-, 4- and 5-grams:

1. *Mod<sub>X</sub>* and *Mody*
2. *Mod<sub>A</sub>* *Mod<sub>B</sub>* and *Mod<sub>X</sub>* *Mody*
3. *Mod<sub>A</sub>* *Mod<sub>B</sub>* and *Mod<sub>X</sub>*
4. *Mod<sub>X</sub>* and *Mod<sub>A</sub>* *Mod<sub>B</sub>*
5. *Mod<sub>X</sub>* and *Mody* *PluralNoun*

*E.g.*, these patterns find the following equivalences:

1. "Batman and Superman"
2. "James Bond and Austin Powers"
3. "Sin City and Gladiator"
4. "Microsoft and Sun Microsystems"
5. "Playboy and Penthouse magazines"

These patterns show the scope for noise when dealing with isolated n-grams. We might ask, what makes the 4-gram *Sin City and Gladiator* a valid coordination but the 3-gram *City and Gladiator* an invalid one? Quite simply, the latter 3-gram does not

yield a pairing that can be grounded in any pair of complex categories, while the 4-gram yields the analogies *Sin City writer* → *Gladiator writer*, *Sin City director* → *Gladiator director*, and so on. Likewise, the substitution *Apples* and *Oranges* is not sensible for the category *Apple CEO* because the category *Orange CEO* does not make sense.

To summarize then, the process of generating inter-category analogies is both straightforward and lightweight. No external knowledge is needed, e.g., to tell the system that *Playboy* and *Penthouse* are both magazines of a somewhat sordid genre, or that *Batman* and *Superman* are both comic-book superheroes (interestingly, WordNet has entries for all four of these words, but assigns them senses that are utterly distinct from their pop-culture meanings). Rather, we simply use coordination patterns to formulate substitutability hypotheses in the context of existing ontological categories. Thus, if a substitution in one existing category yields another existing category, then these two categories are held to be connected by an analogy. We note that one does not have to use Google n-grams to acquire coordination patterns, but can use any corpus at all, thereby tuning the analogical mappings to the sensibilities of a given corpus/context/domain.

When applied to the complex categories of *NameDropper*, using coordination patterns in the Google n-grams, this approach generates 218,212 analogical mappings for 16,834 different categories, with a mean of 12 analogical mappings per category.

## 5 Analogical Cliques

These analogical mappings provide a high degree of pair-wise connectivity between the complex categories of an ontology like *NameDropper*, or of any ontology where category-labels are linguistically complex and amenable to corpus analysis. This connectivity serves to link instances in ways that extend beyond their own categories. Returning to the *Playboy* example, we see the following mappings:

1. *Playboy publisher* → *Penthouse publisher*
2. *Playboy publisher* → *Hustler publisher*
3. *Hustler publisher* → *Penthouse publisher*

All mappings are symmetric, so what we have here is an analogical clique, that is, a complete sub-graph of the overall graph of analogical mappings. Such cliques allow us to generalize upon the pair-wise connectivity offered by individual mappings, to create tightly-knit clusters of mappings that can act as generalizations for the categories involved. Thus, the above mappings form the following clique:

$$\{\textit{Playboy publisher}, \textit{Penthouse publisher}, \textit{Hustler publisher}\}$$

A corresponding clique of modifiers is also implied:

$$\{\textit{Playboy}, \textit{Penthouse}, \textit{Hustler}\}$$

In turn, an analogical clique of categories also implies a corresponding clique of their instances:

$$\{\textit{Hugh Hefner}, \textit{Bob Guccione}, \textit{Larry Flynt}\}$$

It is worth noting that this clique of individuals (who are all linked in the public imagination) does not actually occur in the cliques of proper-named entities that we earlier extracted from the Google 5-grams in section 2 (see Figure 2). In other words, the analogical clique allows us to generalize beyond the confines of the corpus, to create connections that are implied but not always overtly present in the data.

The cohesiveness of an ontological category finds apt expression in the social metaphor of a clique. No element can be added to a clique unless that new element is connected to all the members of the clique. For instance, since *Playboy* magazine is a rather tame example of its genre, we find it coordinated with other, less questionable magazines in the Google n-grams, such as *Sports Illustrated*, *Vanity Fair*, *Rolling Stone* and *Maxim magazines*. Thus, we also obtain mappings like the following:

$$\textit{Playboy publisher} \rightarrow \textit{Rolling Stone publisher}$$

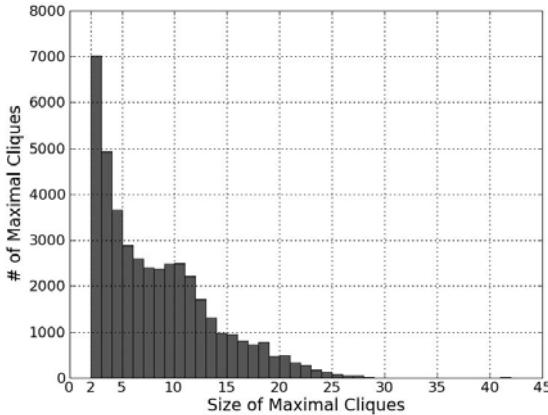
This, in turn, implies a correspondence of instances:

$$\textit{Hugh Hefner} \rightarrow \textit{Jann Wenner}$$

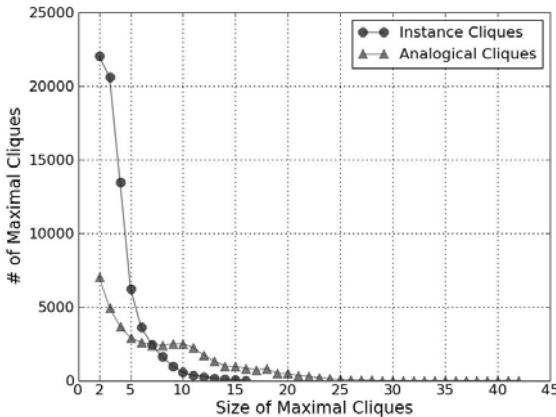
All this is as one might expect, but note how the association of *Playboy* and *Rolling Stone* does not influence the structure of our earlier analogical clique: *Rolling Stone publisher* does not join the clique of *Playboy publisher*, *Penthouse publisher* and *Hustler publisher* because it lacks a connection to the latter two categories; *Jann Wenner* thus avoids membership in the clique of *Hugh Hefner*, *Bob Guccione* and *Larry Flynt*.

Analogical cliques allow us to turn pair-wise analogical mappings between categories into cohesive superordinate categories in their own right. Thus,  $\{\textit{Playboy publisher}, \textit{Penthouse publisher}, \textit{Hustler publisher}\}$  acts as a super-ordinate for the categories *Playboy publisher*, *Penthouse publisher* and *Hustler publisher*, and in turn serves as a common category for *Hugh Hefner*, *Bob Guccione* and *Larry Flynt*. Because analogies are derived in a relatively knowledge-lite manner from corpora, these cliques act as proxies for the kind of explicit categories that a human engineer might define and name, such as *publisher of men's magazines*. Analogical cliques can serve a useful structural role in an ontology without being explicitly named in this fashion, but they can also be extremely useful as part of semi-automated knowledge-engineering solution. In such a system, analogical cliques can be used to find clusters of categories in an ontology for which there is linguistic evidence — as mined from a corpus — for a new super-ordinate category. Once identified in this way, a human ontologist can decide to accept the clique and give it a name, whereupon it is added as a new first-class category to the ontology.

Recall from section 2 that mining the Google n-grams for coordination among proper-named entities yields a highly fragmented set of instance-level cliques. In particular, Figure 2 revealed that clustering instances based on their co-occurrence in corpora produces a very large set of relatively small cliques, rather than the smaller set of larger cliques that one would expect from a sensible categorization scheme. In contrast, Figure 3 below shows that the graph of analogical mappings between categories produces a wider distribution of clique sizes, and produces many more maximal  $k$ -cliques of  $k > 10$ .



**Fig. 3.** Cliques of different sizes from the graph of analogical mappings between *NameDropper* categories



**Fig. 4.** The distribution of instance-level clique sizes (from coordinated proper-names) compared with the distribution of analogical-clique sizes

Figure 4 presents a side-by-side comparison of the results of Figures 2 and 3. It shows that while the analogical level produces less cliques overall (42,340 analogical cliques versus 72,295 instance level cliques, to be specific), analogical cliques tend to be larger in size, and thus achieve greater levels of generalization than cliques derived from instances directly.

Membership in a clique is a rich but implicit source of categorial insights. When two concepts reside in the same clique, this is evidence that they are interchangeable in some contexts, and may even belong to the same conceptual category. To see if clique membership provides enough implicit cues to enable robust category-level reasoning, we replicate the clustering experiments of [16]. These authors take as their test set

214 words from 13 categories in WordNet. Using query patterns in the style of [10] to retrieve informative text fragments for each word from the web, they harvest a large body of features for each word. These features include attributes, such as *temperature* for *coffee*, and attribute values, such as *fast* for *car*. In all, they harvest a set of approx. 60,000 features for the 214 word dataset, and use a standard clustering package to divide the words into 13 clusters on the basis of their web features. They report a cluster purity of 0.85 relative to the original WordNet categories.

We replicate the experiment using the same 214 words and 13 WordNet categories. Rather than harvesting web features for each word, we use the co-members of every maximal clique that the word belongs to. Thus, the features for *chair* are the comparable terms for *chair*, that is, the set of  $X$  such that either “*chairs and Xs*” or “*Xs and chairs*” is a Google 3-gram. We also treat every word as a feature of itself, so e.g., *chair* is a feature of “*chair*”. The sparse matrix produces a much smaller set of features just 8,300 in total for all 214 words. When clustering into 13 groupings using the same clustering software, we obtain a cluster purity of 0.934 relative to WordNet. The space of sensible comparisons, as based on clique co-membership, turns out to be a very compact and precise means of representing the semantic potential of words.

## 6 Conclusions

Word usage in context often defies our best attempts to exhaustively enumerate all the possible senses of a word (e.g., see [17]). Though resources like WordNet are generally very useful for language processing tasks, it is unreasonable to assume that WordNet — or any print dictionary, for that matter — offers a definitive solution to the problem of lexical ambiguity. As we have seen here, the senses that words acquire in specific contexts are sometimes at great variance to the *official* senses that these words have in dictionaries [18]. It is thus unwise to place too great a reliance on dictionaries when acquiring ontological structures from corpora.

We have described here a lightweight approach to the acquisition of ontological structure that uses WordNet as little more than an inventory of nouns and adjectives, rather than as an inventory of senses. The insight at work here is not a new one: one can ascertain the semantics of a term by the company it keeps in a text, and if enough inter-locking patterns are employed to minimize the risk of noise, real knowledge about the use and meaning of words can be acquired [11]. Because words are often used in senses that go beyond the official inventories of dictionaries, resources like WordNet can actually be an impediment to achieving the kinds of semantic generalizations demanded by a domain ontology.

A lightweight approach is workable only if other constraints take the place of lexical semantics in separating valuable ontological content from ill-formed or meaningless noise. In this paper we have discussed two such inter-locking constraints, in the form of clique structures and analogical mappings. Clique structures winnow out coincidences in the data to focus only on patterns that have high internal consistency. Likewise, analogical mappings enforce a kind of internal symmetry on an ontology, biasing a knowledge representation toward parallel structures that recur in many different categories.

## References

1. Gruber, T.: A translation approach to portable ontologies. *Knowledge Acquisition* 5(2), 199–220 (1993)
2. Formal Ontology and Information Systems. In: Guarino, N. (ed.) *Proceedings of FOIS 1998*, Trento, Italy, June 6–8, IOS Press, Amsterdam (1998)
3. Euzenat, J., Shvaiko, P.: *Ontology Matching*. Springer, Heidelberg (2007)
4. De Leenheer, P., de Moor, A.: Context-driven Disambiguation in Ontology Elicitation. In: Shvaiko, P., Euzenat, J. (eds.) *Context and Ontologies: Theory, Practice and Applications*, AAAI Technical Report WS-05-01:17–24. AAAI Press, Menlo Park (2005)
5. Falkenhainer, B., Forbus, K., Gentner, D.: The Structure-Mapping Engine: Algorithm and Examples. *Artificial Intelligence* 41, 1–63 (1989)
6. Veale, T., Li, G., Hao, Y.: Growing Finely-Discriminating Taxonomies from Seeds of Varying Quality and Size. In: *EACL 2009*, Athens (2009)
7. Fellbaum, C. (ed.): *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge (1998)
8. Dong, Z., Dong, Q.: *HowNet and the Computation of Meaning*. World Scientific, Singapore (2006)
9. Budanitsky, A., Hirst, G.: Evaluating WordNet based Measures of Lexical Semantic Relatedness. *Computational Linguistics* 32(1), 13–47 (2006)
10. Hearst, M.: Automatic acquisition of hyponyms from large text corpora. In: *14th International Conference on Computational Linguistics*, pp. 539–545 (1992)
11. Widdows, D., Dorow, B.: A graph model for unsupervised lexical acquisition. In: *19th Int. Conference on Computational Linguistics* (2002)
12. Brants, T., Franz, A.: Web 1t 5-gram version 1. In: *Linguistic Data Consortium*, Philadelphia (2006)
13. Bron, C., Kerbosch, J.: Algorithm 457: Finding all cliques of an undirected graph. *Communications of the ACM* 16(9) (1973)
14. Croitoru, M., Hu, B., Srinandan, D., Lewis, P., Dupplaw, D., Xiao, L.: A Conceptual Graph-based Approach to Ontology Similarity Measure. In: *15th International Conference On Conceptual Structures, ICCS 2007*, Sheffield, UK (2007)
15. Barsalou, L.W.: Ad hoc categories. *Memory and Cognition* 11, 211–227 (1983)
16. Almuhareb, A., Poesio, M.: Attribute-based and Value-based Clustering: An Evaluation. In: *EMNLP*, Barcelona (July 2004)
17. Cruse, D.A.: *Lexical Semantics*. Cambridge University Press, Cambridge (1986)
18. Kilgarriff, A.: I don't believe in word senses. *Computers and the Humanities* 31(2), 91–113 (1997)

# Detection and Transformation of Ontology Patterns

Ondřej Šváb-Zamazal<sup>1</sup>, Vojtěch Svátek<sup>1</sup>, François Scharffe<sup>2</sup>, and Jérôme David<sup>2</sup>

<sup>1</sup> University of Economics, Prague, Czech Republic

{ondrej.zamazal, svatek}@vse.cz

<sup>2</sup> INRIA & LIG, Montbonnot, France

{jerome.david, francois.scharffe}@inrialpes.fr

**Abstract.** As more and more ontology designers follow the pattern-based approach, automatic analysis of those structures and their exploitation in semantic tools is becoming more doable and important. We present an approach to ontology transformation based on transformation patterns, which could assist in many semantic tasks (such as reasoning, modularisation or matching). Ontology transformation can be applied on parts of ontologies called ontology patterns. Detection of ontology patterns can be specific for a given use case, or generic. We first present generic detection patterns along with some experimental results, and then detection patterns specific for ontology matching. Furthermore, we detail the ontology transformation phase along with an example of transformation pattern based on an alignment pattern.

## 1 Introduction

On-demand ontology transformation *inside a formalism* such as OWL<sup>1</sup> can be useful for many semantic applications. The motivation for transformation is that the same conceptualization can be formally modeled in diverse ways; (parts of) an ontology thus can be transformed from one *modeling choice* to another, taking advantage of logical ontology patterns, such as those published by W3C and referring to e.g. ‘n-ary relations’ [5] or ‘specified values’. Although the strictly formal semantics may change, the *intended meaning* of the conceptualisation should be preserved.

In [15] are described three use cases:

- **Reasoning.** Some features of ontologies cause performance problems for certain reasoners. Having information about these features, possibly gathered via machine learning methods, we can transform parts of ontologies with such problematic entities.
- **Modularization.** Modular ontologies are a pre-requisite for effective knowledge sharing, often through *importing*. However, if the source and target ontology are modeled using different styles (such as property- vs. relation-centric), the user faces difficulties when choosing fragments to be imported.
- **Matching.** Most *ontology matching* (OM) tools deliver simple entity-to-entity correspondences. Complex matching can be mediated by alignment (originally called ‘correspondence’) patterns [9], which however most OM tools do not support. Attempting to transform, prior to matching, an ontology to its variant using transformation patterns, could thus help the OM tools.

---

<sup>1</sup> <http://www.w3.org/TR/owl2-primer/>

These use cases share an *ontology transformation service* that makes use of *ontology transformation patterns*. In this paper we present details about our pattern detection and transformation approach. We propose generic detection based on SPARQL query and sketch the specific detection within ontology matching context. Ontology transformation is based on transformation patterns which are close to alignment patterns.

The rest of the paper is organized as follows. Section 2 presents the overall workflow of ontology transformation, followed with an illustrative example. Section 3 details the generic ontology pattern detection along with results of experiment. Next, we describe specific detection method in context of ontology matching. Section 4 presents the relationship between transformation pattern and alignment pattern. The paper is wrapped up with Related work, Conclusions and Future Work.

The paper is an updated and an extended version of [17]; the major novelty is in the implemented RESTful services and in pattern detection method that takes account of the target ontology and applies clustering to the pre-matched correspondences.

## 2 Ontology Transformation Process

### 2.1 Workflow of Ontology Transformation

This section presents the workflow of the ontology transformation. The transformation as such takes as input an ontology  $O_1$  and an ontology transformation pattern. It outputs a new ontology  $O'_1$  resulting from applying an ontology transformation pattern on  $O_1$ . An ontology transformation pattern consists of an ontology pattern  $A$ , its counterpart ontology pattern  $B$ , and a pattern transformation between them.

**Definition 1 (Ontology Pattern).** Ontology Pattern *consists of*:

- Mandatory non-empty set  $E$  of entity declarations, i.e. axioms with `rdf:type` property, in which entity placeholders are used instead of concrete entities.<sup>2</sup>
- Optional set  $Ax$  of axioms asserting facts about entities from  $E$ .
- An optional naming pattern  $NP$  capturing the naming aspect of the ontology pattern relevant for its detection.

**Definition 2 (Pattern Transformation).** Pattern Transformation ( $PT$ ) *consists of*:

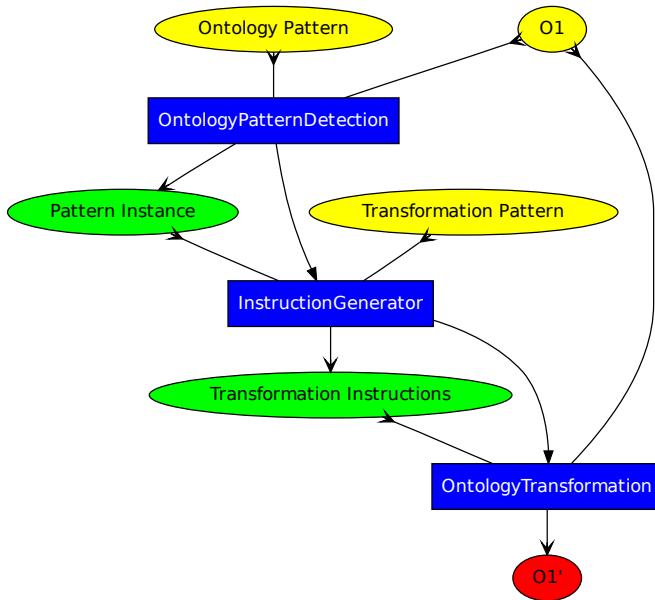
- Mandatory set  $LI$  of links, where a link can be either an equivalence correspondence<sup>3</sup> or an extralogical link `eqAnn` between an annotation literal and a real entity<sup>4</sup> or a link `eqHet` between heterogeneous entities.
- Optional set  $ENP$  of entity naming transformation patterns.

---

<sup>2</sup> Placeholders are used in transformation patterns in general.

<sup>3</sup> Currently, we do not consider further correspondence relations such as `disjointWith`, `subClassOf` etc. for specifying specific relation between old and new version of entity (i.e. versioning). It can be considered in future work.

<sup>4</sup> An annotation literal is an entity that is only used for annotation purposes, while a real entity is real in this sense.



**Fig. 1.** Ontology Transformation Workflow

The ontology pattern  $A$  and the ontology pattern  $B$  typically represent the same conceptualization modeled in two different ways. The transformation pattern captures information on which entities should be transformed and how. This is similar to alignment patterns to some extent, see Section 4.

In Figure 1 you can see this three-step workflow of ontology transformation (for more details see [16]). Rectangle-shaped boxes represent RESTful services, while ellipse-shaped boxes represent input/output data.<sup>5</sup> Ontology transformation is broken up into three basic services:<sup>6</sup>

The *OntologyPatternDetection* service<sup>7</sup> outputs the binding of placeholders in XML. It takes the transformation pattern with its ontology patterns and particular ontology on input. This service internally automatically generates a query based on the ontology pattern and executes it. The structural/logical aspect is captured as a SPARQL query, and the naming constraint is specifically dealt with based on its description within the ontology pattern. The service has been partly implemented by now.<sup>8</sup> Details about the two kinds of detection are in Section 3.

<sup>5</sup> In colours, blue boxes represent RESTful services; yellow ones represent input; green ones represent output which are in next step input; red ones represent output.

<sup>6</sup> Accessible via an HTML user interface at <http://owl.vse.cz:8080/>

<sup>7</sup> <http://owl.vse.cz:8080/ontologyTransformation/detection/>

<sup>8</sup> Its full implementation will leverage on the support of Manchester syntax in SPARQL, which will be available in the new release of Pellet reasoner at the end of March, 2010.

The *InstructionGenerator* service<sup>9</sup> outputs particular transformation instructions in XML. It takes the particular binding of placeholders and the transformation pattern<sup>10</sup> on input. Transformation instructions are generated according to the transformation pattern and the pattern instance.

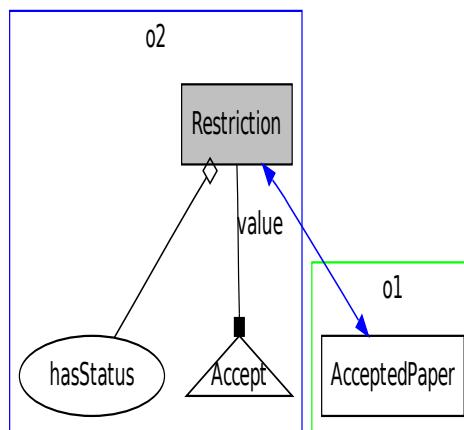
The *OntologyTransformation* service<sup>11</sup> outputs the transformed ontology. It takes particular transformation instructions and the particular ontology on input. This service is partly based on the *Ontology Pre-Processor Lanaguage* (OPPL) [3] and partly on our specific implementation based on the OWL-API.<sup>12</sup>

All these services are implemented as RESTful services available via POST requests. There is also an aggregative one-step service<sup>13</sup> that takes the source ontology, transformation pattern and pattern instance on input and returns the transformed ontology.

During the ontology transformation process, many ontology transformation patterns from the input library may be detected and applied. Moreover, an ontology transformation pattern may be applied many times if its ontology pattern  $A$  was repeatedly detected in the ontology.

## 2.2 Example of Ontology Transformation

In this section we provide an example of transformation pattern based on an alignment pattern. We can imagine the situation where in one ontology the notion of ‘accepted paper’ is formalized through an existential restriction while in the other ontology it is a named class. This corresponds to the ‘Class By Attribute Value’ Pattern, depicted in Figure 2.



**Fig. 2.** Instance of alignment pattern ‘Class By Attribute Value’

<sup>9</sup> <http://owl.vse.cz:8080/ontologyTransformation/instructions/>

<sup>10</sup> Note that the ontology pattern is part of the transformation pattern.

<sup>11</sup> <http://owl.vse.cz:8080/ontologyTransformation/transformation/>

<sup>12</sup> <http://owlapi.sourceforge.net/>

<sup>13</sup> <http://owl.vse.cz:8080/ontologyTransformation/service/>

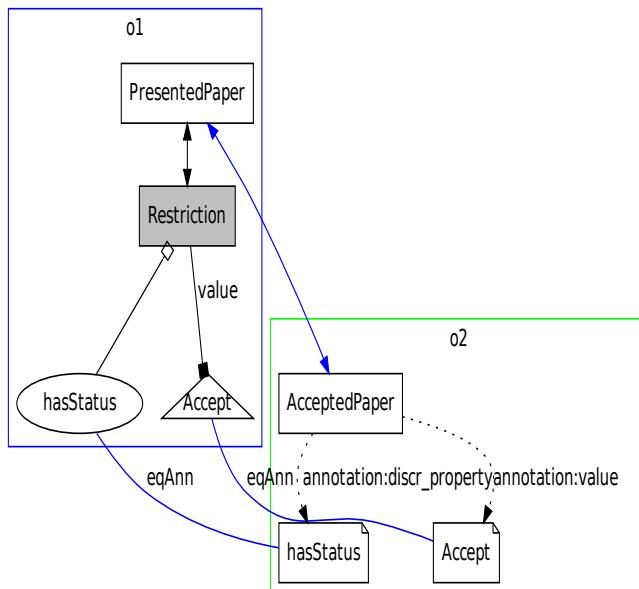
Based on this alignment pattern, a transformation pattern called *tp-hasValue* can be defined as follows:

- *OP1* : E={Class: A, ObjectProperty: p, Individual: a}, Ax={A EquivalentTo: (p values a)}
- *OP2* : E={Class: B, Literal: An1, Literal: An2}, Ax2 = {B annotation: descr\_property An1, B annotation: value An2}
- *PT* : LI={A EquivalentTo: B, p eqAnn: An1, a eqAnn: An2}, enp(B) = make\_passive\_verb(a) + head\_noun(A)

The last item in *PT*, enp(B), is an example of entity naming transformation pattern, which enables to make a proper name for a new entity, e.g. from 'Accept' as *a* and 'PresentedPaper' as *A* it makes 'AcceptedPaper' as *B*.

A particular instantiation of this pattern could be as follows(see Figure 3):

- *O1*: A = *PresentedPaper*, p = *hasStatus*, a = *Accept*
- *O2* : B = *AcceptedPaper*, An1 = ' *hasStatus*' , An2 = ' *Accept*'



**Fig. 3.** Instance of transformation pattern for *hasValue*

By applying the *tp-hasValue* pattern on *O1* we get a new entity 'AcceptedPaper' in *O1'*, which is trivially matchable with the entity 'AcceptedPaper' from *O2*. Besides the new entity 'AcceptedPaper', an annotation is further added that captures the information 'squeezed out' from the logical representation of *O1*; this enables reverse transformation.<sup>14</sup> We can use the alignment pattern included in this transformation pattern in order

<sup>14</sup> There are different strategies how to cope with removing entities and axioms from the original ontology, see [16].

to obtain a complex correspondence, namely: ( $O_1 \# \text{hasStatus} \text{ value } O_1 \# \text{Accept} = O_2 \# \text{AcceptedPaper}$ ).

## 3 Ontology Pattern Detection

### 3.1 Generic Variant

The generic variant of ontology pattern detection (originally introduced in [12]) does not consider the final application of ontology transformation. This phase takes as input the ontology pattern  $A$  of an ontology transformation pattern and tries to match this ontology pattern in the ontology  $O_1$ . The detection of these patterns has two aspects: the structural one and the naming one.

Our method first detect the structural aspect using the SPARQL language.<sup>15</sup> We currently use the SPARQL query engine from the Jena framework;<sup>16</sup> SPARQL queries corresponding to each detected pattern are detailed in sections below.

Then the method applies a lexical heuristic consisting in computing the ratio of the number of distinct tokens shared between the central entity of the pattern (*MainEntity*) and other entities (*Entities*) to the number of all distinct tokens of *Entities*. The particular instantiation of *MainEntity* and *Entities* depends on the ontology pattern, see below. This heuristic works on names of entities (as end fragments of the entity URIs), which are tokenised (see [13]) and lemmatized.<sup>17</sup> Lemmatization can potentially increase the recall of the detection process. The lexical heuristic constraint is fulfilled when the ratio is higher than a certain threshold that is dependent on the particular ontology pattern. This procedure is based on an assumption that entities involved in patterns share tokens: the more entities share the same token, the higher is the probability of occurrence of the pattern.

We detail below three patterns that were detected in the experiment described in Section 3.1.

*Attribute Value Restriction (AVR)*. The AVR pattern has been originally introduced in [9] as a constituent part of an *alignment pattern*, a pattern of correspondence between entities in two ontologies. Basically, it is a class the instances of which are restricted with some attribute value. The SPARQL query for detection of this ontology pattern is the following:

```
SELECT ?c1 ?c2 ?c3
WHERE {
?c1 rdfs:subClassOf _:b.
_:b owl:onProperty ?c2.
_:b owl:hasValue ?c3.
?c2 rdf:type owl:ObjectProperty.
FILTER (!isBlank(?c1)) }
```

<sup>15</sup> <http://www.w3.org/TR/rdf-sparql-query/>

<sup>16</sup> <http://jena.sourceforge.net/>

<sup>17</sup> We use the Stanford POS tagger, <http://nlp.stanford.edu/software/tagger.shtml>

In this query we express a value restriction applied on a named class. Here the restriction class is a superclass of this named class; however we could also employ equivalence as in transformation pattern, see Section 2.2. Furthermore, the restricting properties must be of type 'ObjectProperty' in order to have individuals and not data types as values. Currently we do not consider the naming aspect for this pattern.

*Specified Values (SV).* We first considered the SV pattern in [14]; it had been originally proposed by the W3C SWBPD group.<sup>18</sup> This ontology pattern deals with 'value partitions' representing specified collection of values expressing 'qualities', 'attributes', or 'features'. An example is given in the next section 3.1.

There are mainly two ways of capturing this pattern, which are reflected by two different SPARQL queries. Either individuals where qualities are instances can be used for the detection:

```
SELECT distinct ?p ?a1 ?a2
WHERE {
?a1 rdf:type ?p.
?a2 rdf:type ?p.
?a1 owl:differentFrom ?a2 }
```

or subclasses where qualities are classes partitioning a 'feature' can be used:

```
SELECT distinct ?p ?c1 ?c2
WHERE {
?c1 rdfs:subClassOf ?p.
?c2 rdfs:subClassOf ?p.
?c1 owl:disjointWith ?c2
FILTER (
!isBlank(?c1) && !isBlank(?c2) && !isBlank(?p))}
```

We are interested in mutually disjoint named classes (siblings) and we use the non-transitive semantics of the 'subClassOf' relation (i.e. that of 'direct subclass') here.<sup>19</sup> Regarding the initialization of variables from Algorithm 1, *MainEntity* is either an instance (for the first query, that identified by variable ?p) or class (for the second query). *Entities* are all other entities from the *SELECT* construct. The threshold was experimentally set to 0.5.

*Reified N-ary Relations.* We have already considered the N-ary pattern in [14]. It has also been an important topic of the SWBPD group [5], because there is no direct way how to express N-ary relations in OWL.<sup>20</sup> Basically, a N-ary relation is a relation connecting an individual to more than two individuals or values. For this pattern we adhere to the solution presented in [5]: introducing a new class for a relation, which is thus 'reified'. For examples in the next section 3.1 we will use the following syntax (property(domain,range)):

$$\text{relation}X(X, Y); \text{relation}Y1(Y, A); \text{relation}Y2(Y, B)$$

<sup>18</sup> <http://www.w3.org/TR/swbp-specified-values/>

<sup>19</sup> Otherwise we would get a 'specified value' as many times as there are different superclasses for those siblings.

<sup>20</sup> Not even in version OWL 2.

The structural aspect of this pattern is captured using the following SPARQL query:

```
SELECT ?relationX ?Y ?relationY1 ?relationY2 ?A ?B
WHERE {
?relationX rdfs:domain ?X.
?relationX rdfs:range ?Y.
?relationY1 rdfs:domain ?Y.
?relationY1 rdfs:range ?A.
?relationY2 rdfs:domain ?Y.
?relationY2 rdfs:range ?B
FILTER (?relationY1!=?relationY2) }
```

These conditions (as one variant of detection) are not completely in correspondence with real constraint 'the reified relation class being in the range of one property'. This SPARQL query is rather one experimental way how we tried to detect this pattern. It would be worth trying other more flexible options. In order to increase the precision of the detection, we also apply the lexical heuristic introduced above, where the variable *MainEntity* is initialised with the value *?relationX*. *Entities* are all other entities from the *SELECT* construct. The threshold was experimentally set to 0.4.

*Experiment.* In order to acquire a high number of ontologies, we applied the Watson tool<sup>21</sup> via its API. We searched for ontologies satisfying the conjunction of the following constraints: OWL as the representation language; at least 10 classes; and at least 5 properties. Altogether we collected 490 ontologies. However, many ontologies were not accessible at the time of querying, and for some there were parser problems. Furthermore we only included ontologies having less than 300 entities. All in all our collection has 273 ontologies.

Table 1 presents the overall numbers of ontologies where a certain amount of ontology pattern occurrences was detected.

**Table 1.** Frequency table of ontologies wrt. number of ontology patterns detected.

	$\geq 10$	(9 - 4)	3	2	1	all
AVR pattern	4	-	2	1	1	8
SV pattern	-	4	-	2	9	15
N-ary pattern	-	5	4	16	25	50

We can see that patterns were only detected in a small portion of ontologies from the collection. In four ontologies the AVR pattern was detected more than 10 times. This reflects the fact that some designers tend to extensively use this pattern. The other two ontology patterns were not so frequent in one ontology (the SV pattern was detected maximally 8 times and the N-ary pattern was detected maximally six times). On the other hand the most frequent pattern in terms of number of ontologies was the N-ary pattern. This goes against the intuition that this pattern is rare.

In the following three sections we present three detected positive examples of instances of a given pattern.

<sup>21</sup> [http://watson.kmi.open.ac.uk/WS\\_and\\_API.html](http://watson.kmi.open.ac.uk/WS_and_API.html)

*AVR Pattern.* This ontology pattern was found many times in a wine ontology.<sup>22</sup> One positive example is the following:

$$\text{Chardonnay} \sqsubseteq \exists \text{hasColor.}\{\text{White}\}$$

Chardonnay wine is restricted on these instances having value 'White' for the property hasColor.

*SV Pattern.* The following<sup>23</sup> is one example which we evaluated as positive (a shared token is 'Molecule',  $c = 1.0$ ):

$$\text{AnorganicMolecule} \sqsubseteq \text{Molecule}; \text{OrganicMolecule} \sqsubseteq \text{Molecule}$$

This can be interpreted as a collection of different kinds of molecules as a complete partitioning. Furthermore, disjointness is ensured by a query.

*N-ary Pattern.* Due to the usage of a relaxed structural condition there are a lot of negative cases. Even if the lexical heuristics constraint improves this low precision, there is still ample space for improvement.

In the *PML* ontology<sup>24</sup> the following positive example was detected:

```
hasPrettyNameMapping(InferenceStep, PrettyNameMapping)
hasPrettyName(PrettyNameMapping, string)
hasReplacee(PrettyNameMapping, string)
```

This is an example of N-ary relation where the reified property 'PrettyNameMapping' ('?Y') captures additional attributes ('hasReplacee') describing the relation ('hasPrettyNameMapping'). The threshold was set to  $c = 0.5$ ; shared tokens were 'Pretty' and 'has', respectively.

Once an ontology pattern is detected, the corresponding transformation can be applied as exemplified in Section 2.2.

### 3.2 Specific Variant within Ontology Matching Context

Every use case imposes specific requirements on transformation which should be reflected in the phase of detection in order to improve pattern detection before applying the transformation. In the case of ontology matching, the detection process should consider both to-be-matched ontologies. The approach<sup>25</sup> will be illustrated on two tiny fragments of ontologies,<sup>26</sup>  $O_1$  and  $O_2$ . In real cases, the ontologies would of course contain many more entities. i.e. clustering (explained later) would be more meaningful.

$O_1$ :  $\text{PresentedPaper} \sqsubseteq \text{Paper} \sqsubseteq \text{Document} \equiv \text{Paper} \sqcap \exists \text{hasStatus.}\{\text{Accept}\}$   
 $O_2$ :  $\text{AcceptedPaper} \sqsubseteq \text{Paper} \sqsubseteq \text{Document}$

We start from a set of equivalence correspondences  $\langle O_1 : e_i, O_2 : e_j, = \rangle$  created based on an easy-to-compute lexical distance  $d_L(O_1 : e_i, O_2 : e_j)$ . Using, say, the

<sup>22</sup> <http://www.w3.org/TR/2003/CR-owl-guide-20030818/wine>

<sup>23</sup> <http://www.meteck.org/PilotPollution1.owl>

<sup>24</sup> <http://inferenceweb.stanford.edu/2004/07/iw.owl>

<sup>25</sup> Initially presented in [15].

<sup>26</sup> Inspired by *OntoFarm*, <http://nb.vse.cz/~svabo/oaei2009>

Jaccard measure for  $d_L$ , and filtering out the correspondences below the threshold of 0.5, we get the following six correspondences:

$$\begin{aligned} A &= \langle O1 : Paper, O2 : Paper \rangle, B = \langle O1 : Paper, O2 : AcceptedPaper \rangle, \\ C &= \langle O1 : PresentedPaper, O2 : AcceptedPaper \rangle, D = \langle O1 : Accept, O2 : AcceptedPaper \rangle, \\ E &= \langle O1 : PresentedPaper, O2 : Paper \rangle, F = \langle O1 : Document, O2 : Document \rangle. \end{aligned}$$

Second, these correspondences are clustered based on the aggregation—here, average—of structural distances of entities (for each of  $O1, O2$  separately) involved in them:  $d(c_i, c_j) = avg(d_S(O1 : e_i, O1 : e_j), d_S(O2 : e_i, O2 : e_j))$  where  $c_i$  and  $c_j$  are correspondences between  $O1$  and  $O2$ , and  $d_S$  is a structural distance computed as the minimal number of edges (i.e. an edge is any kind of property relating two entities) between the entities. For example,  $d_S(O1 : Paper, O1 : Accept) = 2$  and  $d_S(O2 : Paper, O2 : AcceptedPaper) = 1$ . The initial matrix of distances between correspondences used for their clustering is in Table 2. Edge counting could of course be replaced with more elaborate ontology distance measuring, as in [2].

**Table 2.**

	A	B	C	D	E
A	-	-	-	-	-
B	0.5	-	-	-	-
C	1.0	0.5	-	-	-
D	1.5	1.0	0.5	-	-
E	0.5	1.0	0.5	1.0	-
F	1.0	1.0	2.0	2.5	1.5

Using hierarchical clustering we might possibly get five out of the six correspondences in the same cluster,  $A, B, C, D, E$ , in which the average distance of correspondences is 1. The output of this phase is the set of entities from  $O1$  taken from this cluster:  $\{Paper, PresentedPaper, Accept\}$ . These entities would be input for the next phase, where patterns would be detected over those entities and corresponding transformation carried out.

## 4 Transformation Patterns vs. Alignment Patterns

As we have already mentioned, *transformation patterns* consist of three components (more details are in [16]): ontology pattern  $A$ , ontology pattern  $B$ , a pattern transformation between them. Pattern transformation consists of links between entities in order to depict which entity from ontology pattern  $A$  should be transformed to which entity from ontology pattern  $B$ . The link can be a logical equivalence correspondence or an extralogical link relating two different kinds of entities, e.g. a property and a class. There is further important information about how to name the newly added entity or how to rename an old entity.

Transformation patterns can be based on matching/alignment patterns [9] considering its equivalence correspondences. But there are important differences in other

aspects. Regarding the purpose, while matching patterns are meant to representat recurring alignment structures at the ontological level,<sup>27</sup>, transformation patterns express how one structure can be transformed to another, conceptually similar structure.

Furthermore, analogously to a correspondences within an alignment pattern there is a pattern transformation part in a transformation pattern. In this part there are transformation links between entities. These links can be defined between homogeneous entities (equivalence correspondences), heterogeneous entities (*eqHet*) and between real and annotation literals (*eqAnn*). These extralogical links enable us to link logical patterns in terms of their modeling alternatives.

Regarding transformation as such, transformation operations are defined over atomic entities: these are renaming, adding and removing operations over axioms, applicable on the source entities. Complex expressions are also considered within a transformation pattern; however, in comparison with alignment patterns they are only meaningful as part of some axiom. It does not make sense to add/remove an unnamed entity (e.g. a restriction class) unless it is involved in some axiom. This means that in the case of matching we consider as matchable components atomic entities and/or (even unnamed) complex expressions. On the other hand, in the case of transformation we only consider as transformable components atomic entities and axioms as wholes.

## 5 Related Work

For the topic of ontology pattern detection, there are two directions of possible related work: those of ontology patterns representation and patterns detection.

Regarding patterns in ontologies in general, the ontology patterns presented here are based on results of the *Semantic Web Best Practices and Deployment Working Group*<sup>28</sup> (SWBPD). There are further activities in this respect like *ontology design patterns* (ODP)<sup>29</sup>. The SWBPD concentrated on *logical patterns*, which are domain-independent, while the ODP portal considers diverse kinds of ontology design patterns (incl. logical patterns, content patterns, reasoning patterns etc.). So far we did not directly reuse patterns from the ODP portal, but we plan to do so in the close future.

While the purpose of these two activities is to provide ontology designers with the best practices on how to model certain situations, we are interested in detecting these ontology patterns. Ontology patterns can emerge by chance, or they can be used intentionally by the ontology designer. In the latter case, detection of ontology patterns should be easier. In both cases, however, as we have ontology transformation in mind, we always take an ontology pattern and its one or more alternatives.

In [8] the authors generally consider using SPARQL expressions for extracting Content Ontology Design Patterns from an existing reference ontology. It is followed by manual selection of particular useful axioms towards creating a new Content Ontology Design Pattern.

---

<sup>27</sup> We restrict all possible alignment patterns to those modeled merely at the ontological level. Alignment patterns driven by data migration are currently omitted in our transformation.

<sup>28</sup> <http://www.w3.org/2001/sw/BestPractices/>

<sup>29</sup> See e.g. <http://ontologydesignpatterns.org>

SPARQL enables us to match structural aspects of ontology patterns by specifying a graph pattern with variables. However, SPARQL is merely a query language for RDF. Ontology patterns, in turn, are DL-like conceptualizations. Therefore we have to consider a translation step between DL-like conceptualizations and the RDF representations, which is not unique. In order to overcome this kind of issue we could use an OWL-DL aware query language, e.g. SPARQL-DL [11]. However this language does not support some specific DL constructs (restrictions) and it has even not been fully implemented yet. Next, we should consider not only asserted axioms but also hidden ones. This could be realized using a reasoner, which could materialize all hidden axioms. Furthermore the SPARQL language is not sufficient for specific lexical constraints (such as synonymy or hyperonymy). We need to either make some additional checking (post-processing), or alternatively to implement a specific SPARQL FILTER function. Such FILTER functions could make the SPARQL language quite expressive; however they are usually computationally expensive [7]. This raises a question of right balance between the expressivity of the query language (here it concerns working with synonyms/hypernyms in a SPARQL query) and computational efficiency. By now, we use a two-phase process for detection of ontology patterns: a SPARQL query for the structural aspects, and then post-processing of the results for lexical constraints.

Regarding the transformation part of our work, an immediate solution would be to use XSLT. However, XSLT transformations are not directly applicable to RDF because of its alternative representations. XSPARQL [1] overcomes this limitation by combining SPARQL with XSLT. XSPARQL constitutes an alternative to detecting and transforming ontology parts as we propose in this paper. It however mixes the detection and transformation parts. As already mentioned in the introduction, we try to keep a clear distinction between the pattern detection and the transformation processes.

Furthermore, there is a large amount of research in *ontology transformation*. It can be divided into transformation within a language (especially OWL) and transformation across languages.

In [10] the authors consider *ontology translation* from the Model Driven Engineering perspective. They concentrate on the way how to represent the alignment as translation rules (essentially on the data level). They argue that it is important to retain clarity and accessibility so as to enable the modellers to view translation problems from three aspects: semantic, lexical and syntactic. Their approach is close to the Unified Modeling Language (UML), they also base their particular text syntax of transformation rules on the associated Atlas Transformation Language (ATL). Input and output patterns are modeled as *MatchedRule*. Furthermore, there are variables (in patterns) and OCL expressions. However, they transfer the ontology translation problem to model driven engineering; they use the MOF model for working with ontologies and translation rules at the data level.

In comparison with the previous work the authors in [4] leverage the ontology translation problem to a generic meta-model. This work has been done from the *model management* perspective, which implies generality of this approach. From this perspective, meta-models are *languages* for defining models. In general, model management tries to ‘support the integration, evolution and matching of (data) models at the conceptual and logical design level’. In comparison with our approach there are important

differences. On the one hand, they consider transformations of ontologies (expressed in OWL DL), but these transformations are considered into generic meta-model or into any other meta-model (ModelGen operator). Their approach is based on the meta-model level from which each meta-model could benefit in the same way, e.g. matching, merging, transforming models between meta-models. On the contrary, in our approach we stay in one meta-model, the OWL language, and we consider transformation as a way of translating a certain representation into its alternatives.

In our current approach we base on OPPL [3], which is a macro language, based on Manchester OWL syntax, for manipulating ontologies written in OWL at the level of axioms. [16] describes how our approach uses this language.

A lot of attention has also been paid to transformation between different modeling languages, transformation based on meta-modeling using UML, and specifically transformation of data-models [6].

## 6 Conclusions and Future Work

In this paper we presented the workflow of pattern-based ontology transformation along with its RESTful services. Furthermore, we presented a generic approach to ontology pattern detection as well as a specific variant for the ontology matching setting. Finally, we presented transformation patterns and their relationships to alignment patterns along with an illustrative example.

In the future we should improve the performance of ontology pattern generic detection as well as do more experiments with detection specific to the ontology matching context. The presented specific detection approach is rather naive. In a real setting, it would probably suffer from the dependency on the initial string-based matching (used merely as trigger for complex matching rather than for ‘equivalence matching’ on its own). In reality, each cluster would also have to be examined for possible *inclusion of nearby entities*, e.g. by checking synonymy using a thesaurus.

Furthermore, we will consider ontology transformation from a system viewpoint, incl. user interaction, incremental selection of patterns, consistency checking of the newly created ontology etc. Furthermore, we will work on extension of the ontology transformation pattern library with other ontology patterns, e.g. name patterns and other patterns based on the ODP portal.

**Acknowledgements.** This work has been partially supported by the CSF grant P202/10/1825 (PatOMat project). The authors would also like to thank Jérôme Euzenat for helpful discussions during early phases of this work and Luigi Iannone for his support about OPPL usage.

## References

1. Akhtar, W., Kopecký, J., Krennwallner, T., Polleres, A.: XSPARQL: Traveling between the XML and RDF worlds – and avoiding the XSLT pilgrimage. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 432–447. Springer, Heidelberg (2008)

2. David, J., Euzenat, J.: Comparison between ontology distances (Preliminary results). In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 245–260. Springer, Heidelberg (2008)
3. Iannone, L., Egana, M., Rector, A., Stevens, R.: Augmenting the Expressivity of the Ontology Pre-Processor Language. In: Proceedings of OWLED 2008 (2008)
4. Kensche, D., Quix, C., Chatti, M.A., Jarke, M.: GeRoMe: A Generic Role Based Metamodel for Model Management. Journal on Data Semantics (2007)
5. Noy, N., Rector, A.: Defining n-ary relations on the semantic web (April 2006)
6. Omelayenko, B., Klein, M. (eds.): Knowledge Transformation for the Semantic Web. IOS press, Amsterdam (2003)
7. Pérez, J., Arenas, M., Gutierrez, C.: Semantics and Complexity of SPARQL. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 30–43. Springer, Heidelberg (2006)
8. Presutti, V., Gangemi, A.: Content ontology design patterns as practical building blocks for web ontologies. In: Li, Q., Spaccapietra, S., Yu, E., Olivé, A. (eds.) ER 2008. LNCS, vol. 5231, pp. 128–141. Springer, Heidelberg (2008)
9. Scharffe, F.: Correspondence Patterns Representation. PhD thesis, University of Innsbruck (2009)
10. Silva Parreiras, F., Staab, S., Schenk, S., Winter, A.: Model driven specification of ontology translations. In: Li, Q., Spaccapietra, S., Yu, E., Olivé, A. (eds.) ER 2008. LNCS, vol. 5231, pp. 484–497. Springer, Heidelberg (2008)
11. Sirin, E., Parsia, B.: SPARQL-DL: SPARQL Query for OWL-DL. In: Proceedings of OWLED 2007 (2007)
12. Šváb-Zamazal, O., Scharffe, F., Svátek, V.: Preliminary results of logical ontology pattern detection using sparql and lexical heuristics. In: Proceedings of WOP 2009 (2009)
13. Šváb-Zamazal, O., Svátek, V.: Analysing Ontological Structures through Name Pattern Tracking. In: Gangemi, A., Euzenat, J. (eds.) EKAW 2008. LNCS (LNAI), vol. 5268, pp. 213–228. Springer, Heidelberg (2008)
14. Šváb-Zamazal, O., Svátek, V.: Towards Ontology Matching via Pattern-Based Detection of Semantic Structures in OWL Ontologies. In: Proceedings of the Znalosti Czechoslovak Knowledge Technology Conference (2009)
15. Šváb-Zamazal, O., Svátek, V., David, J., Scharffe, F.: Towards Metamorphic Semantic Models. In: Poster session at ESWC 2009 (2009)
16. Šváb-Zamazal, O., Svátek, V., Iannone, L.: Pattern-Based Ontology Transformation Service as OPPL Extension. Working draft (2010)
17. Šváb-Zamazal, O., Svátek, V., Scharffe, F.: Pattern-based Ontology Transformation Service. In: KEOD 2009 (2009)

# Islands and Query Answering for Alchi-ontologies

Sebastian Wandelt and Ralf Möller

Hamburg University of Technology, Institute for Software Systems  
Schwarzenbergstr. 95, 21073 Hamburg, Germany  
`{wandelt,r.f.moeller}@tuhh.de`  
<http://www.sts.tu-harburg.de>

**Abstract.** The vision of the Semantic Web fostered the interest in reasoning over ever larger sets of assertional statements in ontologies. Today, real-world ontologies do not fit into main memory anymore and therefore tableaux-based reasoning systems cannot handle these large ontologies any longer.

We propose strategies to overcome this problem by performing query answering for an ontology over (usually small) relevant subsets of assertional axioms, called islands. These islands are computed based on a partitioning-criteria. We propose a way to preserve the partitions while updating an ontology and thus enable stream like reasoning for description logic ontologies. Furthermore, we explain how islands can be used to answer grounded conjunctive queries for description logic ontologies. We think that our proposal can support description logic systems to deal with the upcoming large amounts of fluctuant assertional data.

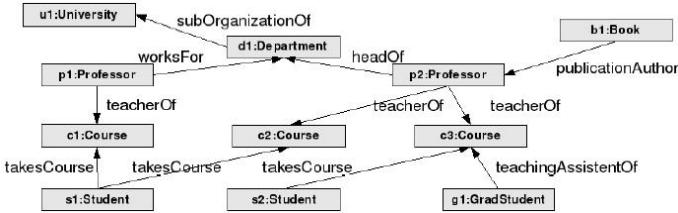
**Keywords:** Description logics, Reasoning, Scalability, Partitioning.

## 1 Introduction

As the Semantic Web evolves, scalability of inference techniques becomes increasingly important. Even for basic description logic-based inference techniques, e.g. instance checking, it is only recently understood on how to perform reasoning on large ABoxes in an efficient way. This is not yet the case for problems that are too large to fit into main memory.

In this paper we present an approach to execute efficient retrieval tests on ontologies, which do not fit into main memory. Existing tableau-based description logic reasoning systems, e.g. Racer [5], do not perform well in such scenarios since the implementation of tableau-algorithms is usually built based on efficient in-memory structures. Our contribution is concerned with the following main objective: we want to partition the assertional part of an *ALCHI*-ontology to more efficiently answer queries over partitions, instead of the complete ABox. The idea is to split up redundant/unimportant role assertions and then partition the ABox based on individual connectedness.

Moreover, we focus on the problem of updating ontologies. The idea is that a partitioning does not need to be computed from the scratch whenever the underlying ontology is changed. To solve that, we propose partitioning-preserving transformations for each possible syntactic update of an ontology (terminological and assertional updates). We are convinced that such an incremental approach is crucial to enable stream-like processing of ontologies.



**Fig. 1.** Guiding Example: ABox  $\mathcal{A}_{EX}$  for ontology  $\mathcal{O}_{EX}$

We propose ways to handle common kinds of queries over description logic ontologies, i.e. instance check queries, instance retrieval queries and grounded conjunctive queries.

The remaining parts of the paper are structured as follows. Section 2 introduces necessary formal notions and gives an overview over Related Work. In Section 3 we introduce the underlying partitioning algorithm, and propose our partitioning-preserving transformations in Section 4 (assertional updates) and in Section 5 (terminological updates). We present our preliminary implementation and evaluation in Section 6. In Section 7, we give insights on query answering over islands. The paper is concluded in Section 8.

## 2 Foundations

### 2.1 Description Logic $\mathcal{ALCHI}$

We briefly recall syntax and semantics of the description logic  $\mathcal{ALCHI}$ . For the details, please refer to [1]. We assume a collection of disjoint sets: a set of *concept names*  $N_{CN}$ , a set of *role names*  $N_{RN}$  and a set of *individual names*  $N_I$ . The set of *roles*  $N_R$  is  $N_{RN} \cup \{R^- | R \in N_{RN}\}$ . The set of  $\mathcal{ALCHI}$ -concept descriptions is given by the following grammar:

$$C, D ::= \top | \perp | A | \neg C | C \sqcap D | C \sqcup D | \forall R.C | \exists R.C$$

where  $A \in N_{CN}$  and  $R \in N_R$ . With  $N_C$  we denote all *atomic concepts*, i.e. concept descriptions which are concept names. For the semantics please refer to [1].

A TBox is a set of so-called *generalized concept inclusions*(GCIs)  $C \sqsubseteq D$ . A RBox is a set of so-called *role inclusions*  $R \sqsubseteq S$ . An ABox is a set of so-called *concept and role assertions*  $a : C$  and  $R(a, b)$ . A ontology  $\mathcal{O}$  consists of a 3-tuple  $\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ , where  $\mathcal{T}$  is a TBox,  $\mathcal{R}$  is a RBox and  $\mathcal{A}$  is a ABox. We restrict the concept assertions in  $\mathcal{A}$  in such a way that each concept description is an atomic concept or a negated atomic concept. This is a common assumption, e.g. in [3], when dealing with large assertional datasets in ontologies.

In the following we define an example ontology, which is used throughout the remaining part of the paper. The ontology is inspired by LUBM [4], a benchmark-ontology in the setting of universities. Although this is a synthetic benchmark, several

(if not most) papers on scalability of ontological reasoning consider it as a base reference. We take a particular snapshot from the LUBM-ontology (TBox, RBox and ABox) and adapt it for presentation purposes. Please note that we do not claim that our snapshot is representative for LUBM.

**Example 1.** Let  $\mathcal{O}_{EX} = \langle \mathcal{T}_{EX}, \mathcal{R}_{EX}, \mathcal{A}_{EX} \rangle$ , s.t.

$$\begin{aligned}\mathcal{T}_{EX} = & \{ \\ & \text{Chair} \equiv \exists \text{headOf}. \text{Department} \sqcap \text{Person}, \text{Professor} \sqsubseteq \text{Faculty}, \\ & \text{Book} \sqsubseteq \text{Publication}, \\ & \text{GraduateStudent} \sqsubseteq \text{Student}, \text{Student} \equiv \text{Person} \sqcap \exists \text{takesCourse}. \text{Course}, \\ & \top \sqsubseteq \forall \text{teacherOf}. \text{Course}, \exists \text{teacherOf}. \top \sqsubseteq \text{Faculty}, \text{Faculty} \sqsubseteq \text{Person}, \\ & \top \sqsubseteq \forall \text{publicationAuthor}^{-}. (\text{Book} \sqcup \text{ConferencePaper}) \\ & \}\end{aligned}$$

$$\mathcal{R}_{EX} = \{\text{headOf} \sqsubseteq \text{worksFor}, \text{worksFor} \sqsubseteq \text{memberOf}, \text{memberOf} \doteq \text{member}^{-}\}$$

$$\mathcal{A}_{EX} = \text{see Figure 1}$$

## 2.2 Related Work

Referring to Example 1, different kinds of partitionings can be, informally, summarized as follows:

- Naive partitioning: This partitioning is done in existing reasoning systems. The idea is that individuals end up in the same partition, if there is a path of role assertions connecting them. Usually many individuals are connected to most other individuals in an ontology. This basic partitioning strategy is often not enough. In our LUBM-example there is only one partition, since each named individual is connected via a path to each other named individual.
- Extension in [3]: Since *suborganizationOf* and *teachingAssistantOf* are the only roles, which are not bound in a  $\forall$ -constraint in  $\mathcal{T}_{EX}$  (please note that *takesCourse* occurs indirectly in a  $\forall$ -constraint when the definition of student is split up into two inclusions), there are three partitions:
  1. one partition containing university  $u1$ ,
  2. one partition containing graduate student  $g1$  and
  3. one partition containing all remaining individuals
- Our proposal: a more fine-grained partitioning (details see below). For example, the only sub-concepts, which can be propagated over the role *teacherOf* are  $\perp$  and *Course*. Now, since for role assertion  $\text{teacherOf}(p1, c1)$ ,  $c1$  is an explicit instance of *Course*, i.e. the propagation is redundant, we can informally speaking “split up” the assertion to further increase granularity of connectedness-based partitioning.

There exists further related work on scalable reasoning. In [2], the authors suggest a scalable way to check consistency of ABoxes. The idea is to merge edges in an ABox whenever consistency is preserved. Their approach is query dependent and, informally speaking, orthogonal to partitioning approaches.

Several papers discuss the transformation of an ontology into datalog, e.g. [7], or the use of novel less-deterministic hypetableau algorithms[8], to perform scalable reasoning. Furthermore, [10] suggests to partition the terminological part of an ontology, while we focus on the assertional part.

After all, we think that our work can be seen as complementary to other work, since it can be easily incorporated into existing algorithms. Furthermore we are unique in focusing on updating partitions to support stream-like processing.

### 3 Ontology Partitioning

We have initially proposed a method for role assertion separability checking in [11]. For completeness we start with one definition from [11]. The definition of  $\mathcal{O}$ -separability is used to determine the importance of role assertions in a given ABox. Informally speaking, the idea is that  $\mathcal{O}$ -separable assertions will never be used to propagate “complex and new information” (see below) via role assertions.

**Definition 1.** *Given an ontology  $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ , a role assertion  $R(a, b)$  is called  $\mathcal{O}$ -separable, if we have  $\mathcal{O}$  is inconsistent  $\iff \langle \mathcal{T}, \mathcal{R}, \mathcal{A}_2 \rangle$  is inconsistent, where*

$$\begin{aligned}\mathcal{A}_2 = \mathcal{A} \setminus \{R(a, b)\} \cup \{R(a, i_1), R(i_2, b)\} \cup \\ \{i_1 : C|b : C \in \mathcal{A}\} \cup \{i_2 : C|a : C \in \mathcal{A}\},\end{aligned}$$

s.t.  $i_1$  and  $i_2$  are fresh individual names.

Now, we further extend our proposal by partitioning-preserving update transformations. To do so, we define a notion of ABox and Ontology partitioning, which will be used in our update transformations below.

**Definition 2.** *Given an ontology  $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ , an ABox Partition for  $\mathcal{A}$  is a tuple  $AP = \langle IN, S \rangle$  such that*

- $IN \subseteq Inds(\mathcal{A})$  and
- $S = \{a : C|a \in M \wedge a : C \in \mathcal{A}\} \cup \{R(a, b)|(a \in IN \vee b \in IN) \wedge R(a, b) \in \mathcal{A}\}$ , where  $M = \{a|b \in IN \wedge (R(a, b) \in \mathcal{A} \vee R(b, a) \in \mathcal{A})\} \cup IN$

We define two projection functions to obtain the first and the second element in a partition-pair: let  $\pi_{IN}(AP) = IN$ , and  $\pi_S(AP) = S$ . Informally speaking, an *ABox Partition* is composed of two components. The individual set  $IN$ , which contains the core individuals of the partition, and the assertion set  $S$  containing all the assertions needed in the partition. If  $a$  is an individual in  $IN$ , then  $S$  contains all the assertions involving  $a$  and all the concept assertions involving all direct neighbours of  $a$ .

**Definition 3.** *Given an ontology  $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ , an ABox Individual Partitioning for  $\mathcal{A}$  is a set  $P = \{ap_1, \dots, ap_n\}$ , such that each  $ap_i$  is an ABox Partition for  $\mathcal{A}$  and*

1. For each  $ap_i, ap_j$ , ( $i \neq j$ ) we have  $\pi_{IN}(ap_i) \cap \pi_{IN}(ap_j) = \emptyset$
2.  $Ind(\mathcal{A}) = \bigcup_{i=1..n} \pi_{IN}(ap_i)$
3.  $\mathcal{A} = \bigcup_{i=1..n} \pi_S(ap_i)$

The definition states that all the partitions have distinct core individual sets, the union of all the core individual sets of all the partitions is exactly the individual set of  $\mathcal{A}$ , and the union of all the assertion sets of all the partitions is the assertion set of  $\mathcal{A}$ .

Since each individual is assigned to only one ABox partition as a core individual, we define a function  $\phi_P : Ind(\mathcal{A}) \rightarrow P$  that returns the partition for a given individual  $a$ . If  $a \notin Ind(\mathcal{A})$ , then  $\phi_P(a) = \emptyset$ . Next we will define the partitioning for the ontology.

**Definition 4.** Given a consistent ontology  $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$ , an Ontology Partitioning for  $\mathcal{O}$  is a structure  $OP_{\mathcal{O}} = \langle \mathcal{T}, \mathcal{R}, P \rangle$ , where  $P$  is an ABox Partitioning for  $\mathcal{A}$  such that for each individual  $a \in Ind(\mathcal{A})$  and each atomic concept  $C$  we have  $\mathcal{O} \models a : C$  iff  $\langle \mathcal{T}, \mathcal{R}, \pi_S(\phi_P(a)) \rangle \models a : C$ .

We use the  $\mathcal{O}$ -separability, see [11], of role assertions to determine the partitioning of  $\mathcal{A}$ . From the previous section, it holds that with the partitioning an ABox based on the  $\mathcal{O}$ -separability of role assertions, the instance checking problem can be solved with only one partition.

## 4 Updating the ABox

In this section, we will introduce means to preserve a partitioning of an ontology under Syntactic ABox Updates[6]. With syntactic updates, there is no consistency checking when adding a new assertion, and neither an enforcement of non-entailment when removing. However, syntactic updates are computationally easier to handle.

The general scenario for updating an ABox is as follows: We assume to start with an empty ontology (which has no assertions in the ABox), and its corresponding partitioning. Then we build up step by step the partitioned ontology by use of our update transformations.

For an empty ontology  $\mathcal{O} = \langle \mathcal{T}, \mathcal{R}, \{\} \rangle$ , the corresponding partitioning is  $OP_{\mathcal{O}} = \langle \mathcal{T}, \mathcal{R}, P \rangle$  where  $P = \{\langle \{\}, \{\} \rangle\}$ . In the following we will use two update functions, *merge* and *reduce*, to implement our update transformations:

**Definition 5.** The result of the merge operation on a set of ABox Partitions for  $\mathcal{A}$ ,  $Merge(\{ap_1, \dots, ap_n\})$ , is defined as the ABox Partition  $ap$  for  $\mathcal{A}$ , s.t.

$$ap = \langle \bigcup_{i \leq n} \pi_{IN}(ap_i), \bigcup_{i \leq n} \pi_S(ap_i) \rangle$$

**Definition 6.** The result of the reduce operation on an ABox Partition for  $\mathcal{A}$ ,  $Reduce(pa)$ , is defined as a set of ABox Partition  $\{ap_1, \dots, ap_n\}$  built as follows:

1. For each  $R(a, b) \in \pi_S(ap)$  do: if  $R(a, b)$  is  $\mathcal{O}$ -separable, then replace  $R(a, b)$  with  $\{R(a, b*), R(a*, b)\} \cup \{a* : C | a : C \in \pi_S(ap)\} \cup \{b* : C | b : C \in \pi_S(ap)\}$ , where  $a*$  and  $b*$  are fresh individual names for  $a$  and  $b$ .
2. Let  $\{ap_1, \dots, ap_n\}$  be the disconnected partitions in  $ap$ .
3. Replace each  $a*$  in each  $ap_i$  by  $a$ .
4. Replace each  $b*$  in each  $ap_i$  by  $b$ .

The *merge* operation simply merges all the core individual sets and the assertion sets of all the partitions. The *reduce* operation, in the other hand, divides an ABox Partition into smaller partitions based on  $\mathcal{O}$ -separability of role assertions.

The algorithm for updating ABoxes is illustrated in Figure 2. It can be informally summarized as follows:

---

Adding a role assertion  $R(a, b)$ :

1. If  $\phi_P(a) = \emptyset$  then add  $\{\{a\}, \{R(a, b)\}\}$  to  $P$
2. If  $\phi_P(b) = \emptyset$  then add  $\{\{b\}, \{R(a, b)\}\}$  to  $P$
3. If  $\phi_P(a) = \phi_P(b)$  then  $\pi_S(\phi_P(a)) = \pi_S(\phi_P(b)) \cup \{R(a, b)\}$
4. Else If  $R(a, b)$  is  $\mathcal{O}$ -separable w.r.t.  $\pi_S(\phi_P(a))$  then
  - (a) Add  $R(a, b)$  to  $\pi_S(\phi_P(a))$  and to  $\pi_S(\phi_P(b))$
  - (b) Add  $\{b : C \mid b : C \in \pi_S(\phi_P(b))\}$  to  $\pi_S(\phi_P(a))$
  - (c) Add  $\{a : C \mid a : C \in \pi_S(\phi_P(a))\}$  to  $\pi_S(\phi_P(b))$
5. Else
  - (a) Add  $R(a, b)$  to  $\pi_S(\phi_P(a))$
  - (b)  $P = P \setminus \{\phi_P(a), \phi_P(b)\} \cup \text{Merge}(\phi_P(a), \phi_P(b))$

---

Removing a role assertion  $R(a, b)$ :

1. If  $\phi_P(a) \neq \phi_P(b)$  then
  - (a)  $\pi_S(\phi_P(a)) = \pi_S(\phi_P(a)) \setminus \{R(a, b)\}$
  - (b)  $\pi_S(\phi_P(b)) = \pi_S(\phi_P(b)) \setminus \{R(a, b)\}$
2. Else
  - (a) If  $R(a, b)$  was  $\mathcal{O}$ -separable w.r.t.  $\pi_S(\phi_P(a))$  then
 
$$\begin{aligned} \pi_S(\phi_P(a)) &= \pi_S(\phi_P(a)) \setminus \{R(a, b)\} \\ \pi_S(\phi_P(b)) &= \pi_S(\phi_P(b)) \setminus \{R(a, b)\} \end{aligned}$$
  - (b) Else  $P = P \setminus \{\phi_P(a), \phi_P(b)\} \cup \text{Reduce}(\text{Merge}(\phi_P(a), \phi_P(b)))$

---

Adding a concept assertion  $a : C$ :

1. If  $\phi_P(a) = \emptyset$  then add  $\{\{a\}, \{\}\}$  to  $P$
2.  $\pi_S(\phi_P(a)) = \pi_S(\phi_P(a)) \cup \{a : C\}$
3. For each  $ap_t \in P$  do
  - If  $a \in Ind(\pi_S(ap_t))$  then  $\pi_S(ap_t) = \pi_S(ap_t) \cup \{a : C\}$
4.  $P = P \setminus \{\phi_P(a)\} \cup \text{Reduce}(\phi_P(a))$

---

Removing a concept assertion  $a : C$ :

1.  $\pi_S(\phi_P(a)) = \pi_S(\phi_P(a)) \setminus \{a : C\}$
2. For each  $ap_t \in P$  do
  - If  $a \in Ind(\pi_S(ap_t))$  then  $\pi_S(ap_t) = \pi_S(ap_t) \setminus \{a : C\}$
3. For each  $R(a, b) \in \phi_P(a)$  do
  - If  $R(a, b) \in \phi_P(b)$  do
    - If  $R(a, b)$  is not  $\mathcal{O}$ -separable, then  $P = P \setminus \{\phi_P(a), \phi_P(b)\} \cup \{\text{Merge}(\phi_P(a), \phi_P(b))\}$
4. For each  $R(b, a) \in \phi_P(a)$  do
  - If  $R(b, a)$  is not  $\mathcal{O}$ -separable, then  $P = P \setminus \{\phi_P(b), \phi_P(a)\} \cup \{\text{Merge}(\phi_P(b), \phi_P(a))\}$

**Fig. 2.** Updating ABox

*Adding a role assertion  $R(a, b)$ :* first we ensure that partitions exist for both  $a$  and  $b$  (if not, create a new partition). If  $a$  and  $b$  are in the same partition, then the role assertion is just simply added to the partition. If  $a$  and  $b$  are in two distinct partitions, and  $R(a, b)$  is not  $\mathcal{O}$ -separable, then the two partitions are merged.

*Removing a role assertion  $R(a, b)$ :* if  $a$  and  $b$  are in different partitions, then the role assertion is just simply removed from both partitions. If  $a$  and  $b$  are in the same partition, then after removing the role assertion the partition needs to be rechecked to see if the removal of the role assertion causes the partition to be reduceable.

*Adding a concept assertion  $C(a)$ :* first we ensure that partition exists for individual  $a$ . Then we add concept assertion  $C(a)$  to the partition of  $a$  ( $\phi_P(a)$ ), and all the partitions that contain any role assertion for  $a$ , to maintain the data consistency between partitions.

*Removing a concept assertion  $C(a)$ :* remove the concept assertion from all the partitions containing it. After that, all the role assertion involving  $a$  need to be  $\mathcal{O}$ -separability checked. If any of the role assertions becomes  $\mathcal{O}$ -inseparable due to the removal, then the corresponding partitions need to be merged.

## 5 Updating the TBox

In the following, we give a rough sketch of the update transformations. For details please refer to our technical report [9]. We extend the definition of the  $\forall$ -info structure from [11], by introducing a *reduced*  $\forall$ -info structure and an *extended*  $\forall$ -info structure.

**Definition 7.** A reduced  $\forall$ -info structure for ontology  $\mathcal{O}$  is a function  $e_{\mathcal{O}}^{\forall}$  which is extend from  $\forall$ -info structure  $f_{\mathcal{O}}^{\forall}$  such that for every role  $R$ :

$$e_{\mathcal{O}}^{\forall}(R) = f_{\mathcal{O}}^{\forall}(R) \setminus \{C_k \mid \exists C \in f_{\mathcal{O}}^{\forall} : C \sqsubset C_k\}$$

**Definition 8.** An extended  $\forall$ -info structure for ontology  $\mathcal{O}$  is a function  $g_{\mathcal{O}}^{\forall}$  which is extended from reduced  $\forall$ -info structure  $e_{\mathcal{O}}^{\forall}$  as following:

- If  $e_{\mathcal{O}}^{\forall}(R) = *$  then  $g_{\mathcal{O}}^{\forall}(R) = \{\langle *, * \rangle\}$
- Else If  $e_{\mathcal{O}}^{\forall}(R) = \emptyset$  then  $g_{\mathcal{O}}^{\forall}(R) = \{\langle \emptyset, \emptyset \rangle\}$
- Else  $g_{\mathcal{O}}^{\forall}(R) = \{\langle C_i, Sub(C_i) \rangle\}$ , with  $C_i \in e_{\mathcal{O}}^{\forall}(R)$ , and  $Sub(C_i)$  is the set of all the concepts that  $C_i$  subsumes in the simple concept hierarchy  $H_S$ .

We also denote  $\pi_C(g_{\mathcal{O}}^{\forall}(R)) \equiv \{C_i\}$ , the set of all  $C_i$  appears in  $\{\langle C_i, Sub(C_i) \rangle\}$  (which is  $e_{\mathcal{O}}^{\forall}(R)$ ); and  $\pi_{Sub,C_i}(g_{\mathcal{O}}^{\forall}(R)) \equiv Sub(C_i)$ .

Informally speaking, the reduced  $\forall$ -info structure contains only the bottommost concepts of the concept hierarchy branches that appears in  $f_{\mathcal{O}}^{\forall}$ , w.r.t. the simple concept hierarchy. On the other hand, an entry in the extended  $\forall$ -info structure is a set, each element of which is a tuples of a concept in  $e_{\mathcal{O}}^{\forall}$  and the set of all the children of that concept, w.r.t. the concept hierarchy.

Updating ABox assertions can lead to the merging/reducing involving one or two specific partitions identified by the individuals in the updated assertions, while updating in TBox and RBox rather causes the merging/reducing in many pairs of partitions involving a certain set of role names. More formally speaking, updating w.r.t TBox and RBox can affects a set of role  $U_R$ , such that for each  $R \in U_R$ , and all individual pairs  $\{a, b\}$ , s.t.  $R(a, b) \in \mathcal{A}$ , the status of the role assertion  $R(a, b)$  might be changed ( $\mathcal{O}$ -separable to  $\mathcal{O}$ -inseparable or vice versa). We call this role set  $U_R$  the *changeable role set*, and each  $R \in U_R$  *changeable role*.

We have derived the following algorithm for updating a TBox and a RBox:

- For each role  $R$  in new terminology  $T*$ , calculate  $g_{\mathcal{O}}^{\forall}(R)$  before updating and  $g_{\mathcal{O}*}^{\forall}(R)$  after updating.
  - If  $(g_{\mathcal{O}}^{\forall}(R) \neq g_{\mathcal{O}*}^{\forall}(R))$  then  $U_R = U_R \cup R$
- For each  $R \in U_R$ , and for each  $R(a, b)$ :
  - If  $R(a, b)$  is  $\mathcal{O}$ -separable but not  $\mathcal{O}*$ -separable then  $P = P \setminus \{\phi_P(a), \phi_P(b)\} \cup Merge(\phi_P(a), \phi_P(b))$
  - If  $R(a, b)$  is not  $\mathcal{O}$ -separable but  $\mathcal{O}*$ -separable then  $P = P \setminus \phi_P(a) \cup Reduce(\phi_P(a))$

(\*)  $\mathcal{O}*$ -separable is denoted for separable with respect to the new ontology (after update), while  $\mathcal{O}$ -separable is denoted for separable with respect to the old ontology.

In the following, we will consider specific cases of updating TBox, and the effects they make to the extended  $\forall$ -info structure, and by this, compute the changeable role set. Then, in case of a terminological update, we have to check all role assertions, whose role is an element of the changeable role set, for  $\mathcal{O}$ -separability.

### 5.1 Updating TBox – Concept Inclusions

Updating TBox by adding/removing a concept inclusion might causes changes to  $g_{\mathcal{O}}^{\forall}$  because

- if the concept inclusion adds  $A \sqsubseteq B$  to the Concept Hierarchy  $H_S$ , and since the extended  $\forall$ -info structure  $g_{\mathcal{O}}^{\forall}$  is built based on  $H_S$ , there probably have changes in  $g_{\mathcal{O}}^{\forall}$ .
- if the SNF, see [11] for details, of the added concept inclusion contains one or more  $\forall$ -bound for a role  $R$  that did not exist in the old terminology (or does not exist in updated terminology in case of removing concept inclusion), then there is changes in the  $\forall$ -info structure of the terminology, which also probably causes changes in the extended  $\forall$ -info structure.

Thus, instead of recalculating the extend  $\forall$ -info structure, if we know that the update is of a concept inclusion, then we just need to extract the information from the added/removed concept inclusion itself to check if it will cause changes in the  $g_{\mathcal{O}}^{\forall}$ .

Before go into details how to decide the update role set from the added/ removed concept inclusion, we introduce some useful definitions.

**Definition 9.** A  $\forall$ -info structure for a concept inclusion  $C \sqsubseteq D$  w.r.t  $\mathcal{O}$ , written as  $f_{C \sqsubseteq D, \mathcal{O}}^{\forall}$ , is a function that assigns to each role name  $R$  in  $\text{SNF}(C \sqsubseteq D)$  one of the following entries:

- $\emptyset$  if we know that there is no  $\forall$  constraint for  $R$  in  $\text{SNF}(C \sqsubseteq D)$ .
- a set  $S$  of atomic concept or negation atomic concept, s.t. there is no other than those in  $S$  that occurs  $\forall$ -bound on  $R$  in  $\text{SNF}(C \sqsubseteq D)$ .
- $*$ , if there are arbitrary complex  $\forall$  constraints on role  $R$  in  $\text{SNF}(C \sqsubseteq D)$ .

This definition is literally similar to the definition of the  $\forall$ -info structure stated before, but for only one axiom. From this, we also define the *reduced  $\forall$ -info structure for a concept inclusion w.r.t. ontology  $\mathcal{O}$*  and *extended  $\forall$ -info structure for a concept inclusion w.r.t. ontology  $\mathcal{O}$*  in the same manner.

**Definition 10.** A reduced  $\forall$ -info structure for a concept inclusion  $C \sqsubseteq D$  w.r.t. ontology  $\mathcal{O}$  is a function  $e_{C \sqsubseteq D, \mathcal{O}}^{\forall}$  which is extend from  $\forall$ -info structure  $f_{C \sqsubseteq D, \mathcal{O}}^{\forall}$  such that for every role  $R$ :

$$e_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R) = f_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R) \setminus \{C_k | \exists C \in f_{C \sqsubseteq D, \mathcal{O}}^{\forall} : C \sqsubseteq C_k\}$$

**Definition 11.** An extended  $\forall$ -info structure for a concept inclusion  $C \sqsubseteq D$  w.r.t. ontology  $\mathcal{O}$  is a function  $g_{C \sqsubseteq D, \mathcal{O}}^{\forall}$  which is extended from reduced  $\forall$ -info structure  $e_{C \sqsubseteq D, \mathcal{O}}^{\forall}$  as following:

- If  $e_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R) = *$  then  $g_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R) = \{(*, *)\}$
- Else If  $e_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R) = \emptyset$  then  $g_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R) = \{(\emptyset, \emptyset)\}$
- Else  $g_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R) = \{\langle C_i, \text{Sub}(C_i) \rangle\}$ , with  $C_i \in e_{C \sqsubseteq D, \mathcal{O}}^{\forall}(R)$ , and  $\text{Sub}(C_i)$  is the set of all the concepts that  $C_i$  subsumes in the simple concept hierarchy  $H_S$ .

And we have the following detailed algorithm for calculating the update role set in case of adding/removing a concept inclusion:

- Adding a concept inclusion  $C \sqsubseteq D$ 
  - For each  $A \sqsubseteq B$  that is added to the concept hierarchy:
    - \* for any role  $R$  that  $B \in g_{\mathcal{O}}^{\vee}(R)$ ,  $U_R = U_R \cup R$
  - For each  $R$  s.t.  $g_{C \sqsubseteq D, \mathcal{O}_*}^{\vee}(R) \neq \emptyset \wedge g_{C \sqsubseteq D, \mathcal{O}_*}^{\vee}(R) \not\subseteq g_{\mathcal{O}}^{\vee}(R)$ ,  $U_R = U_R \cup R$
- Removing a concept inclusion  $C \sqsubseteq D$ 
  - For each  $A \sqsubseteq B$  that is removed to the concept hierarchy:
    - \* for any role  $R$  that  $B \in g_{\mathcal{O}}^{\vee}(R)$ ,  $U_R = U_R \cup R$
  - For each  $R$  s.t.  $g_{C \sqsubseteq D, \mathcal{O}_*}^{\vee}(R) \neq \emptyset \wedge g_{C \sqsubseteq D, \mathcal{O}_*}^{\vee}(R) \not\subseteq g_{\mathcal{O}_*}^{\vee}(R)$ ,  $U_R = U_R \cup R$

Here, we denote with  $\mathcal{O}$  the ontology before updating and with  $\mathcal{O}_*$  the ontology after updating.

## 5.2 Updating RBox – Role Inclusions

Adding/removing a role inclusion has a quite obvious effect: it might change the role hierarchy. Since the  $\forall$ -info structure of the ontology is calculated using role taxonomy, this will change the  $\forall$ -info structure, and also the extended  $\forall$ -info structure. In the following, we present a way to determine the update role set

- Adding a role inclusion  $R \sqsubseteq S$ 
  - if  $g_{\mathcal{O}}^{\vee}(S) \not\subseteq g_{\mathcal{O}}^{\vee}(R)$  then for all sub role  $V$  of  $R$  ( $V \sqsubseteq R$ ),  $U_R = U_R \cup V$
- Removing a role inclusion  $R \sqsubseteq S$ 
  - if  $g_{\mathcal{O}}^{\vee}(S) \not\subseteq g_{\mathcal{O}_*}^{\vee}(R)$  then for all sub role  $V$  of  $R$  ( $V \sqsubseteq R$ ),  $U_R = U_R \cup V$

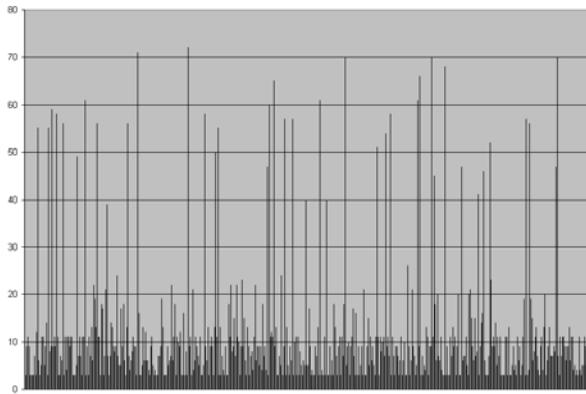
## 5.3 Updating RBox – Role Inverses

Adding/removing a role inverse, on the other hand, might change the  $\forall$ -bound for both roles involving the inverse role. This causes the changes for the  $\forall$ -info structure of both roles, which also alters their extend  $\forall$ -info structure, thus we have following algorithm for calculating update role set:

- Adding a role inverse pair  $R = Inv(S)$ 
  - for all role  $V \sqsubseteq R$ ,  $U_R = U_R \cup V$
  - for all role  $W \sqsubseteq S$ ,  $U_R = U_R \cup W$
- Removing a role inverse pair  $R = Inv(S)$ 
  - for all role  $V \sqsubseteq R$ ,  $U_R = U_R \cup V$
  - for all role  $W \sqsubseteq S$ ,  $U_R = U_R \cup W$

## 6 Distributed Storage System and Preliminary Evaluation

We have implemented the above algorithms in a Java program and performed initial tests on LUBM. The first test is composed of a server and 3 nodes. For the system performance, our test program was able to load 400-500 LUBM-ABox/TBox assertions per second. This is just an average value. From our experience, ABox assertions turn out to be loaded much faster, while TBox assertions slow the system down. The reasons for that behaviour have already been indicated above.



**Fig. 3.** Assertion distribution among partitions in node 1 (3 nodes)

**Table 1.** Partitions and assertions distribution among 3 nodes

Node	Total Partitions	Total Assertions	Assertions/partition	min	max
1	518	6089	11.7548	3	72
2	518	6822	13.1699	3	1596
3	518	5702	11.0077	3	77

**Table 2.** Partitions and assertions distribution among 6 nodes

Node	Total Partition	Total Assertion	Assertion/partition	min	max
1	260	2989	11.4962	3	70
2	259	4129	15.9421	3	1596
3	259	2864	11.0579	3	77
4	258	3100	12.0155	3	72
5	259	2693	10.3977	3	76
6	259	2838	10.9575	3	74

Besides system performance, another factor we want to evaluate is the distribution of the data among nodes. The data collected using three nodes is shown in Table 1. It is easy to see that the number of partitions in the 3 nodes are somehow equally distributed.

Figure 3 illustrates the distribution of the assertions in the partitions on the first node. As shown in the figure, the number of assertions is quite different between partitions. These differences actually illustrate the structure of the test data.

We also ran the testing with four, five and six nodes to collect distribution data. The distribution is somehow similar to the case of 3 nodes. Table 2 listed the data collected for six nodes. The data distribution in our test is somehow nice, with the equally distribution of the partitions among nodes. However, this is the result of some synthetic benchmark data, which does not introduce many merging between partitions. Running our algorithm on more complex data, the partition allocation policy can be a critical factor deciding the system performance.

## 7 Query Answering

In the following section we investigate the problem of query answering over ontologies and in how far our proposal of island partitionings can help to solve problems locally. Solving the problem of *instance checking*, finding out whether  $\mathcal{O} \models a : C$ , is immediate from our proposal of island partitionings. Since we have

$$\mathcal{O} \models a : C \iff \langle \mathcal{T}, \mathcal{R}, \pi_S(\phi_P(a)) \rangle \models a : C,$$

we run a tableaux algorithm on the ontology  $\langle \mathcal{T}, \mathcal{R}, \pi_S(\phi_P(a)) \cup \{a : \neg C\} \rangle$  and check, whether it is consistent. If the ontology is inconsistent, then we proved that  $\mathcal{O} \models a : C$ , and non-entailment otherwise. Thus, instance checking can be performed locally on one node.

To solve the problem of *relation checking* for  $\mathcal{ALCHI}$ , i.e. find out whether  $\mathcal{O} \models R(a, b)$ , we can look at ontology  $\langle \mathcal{T}, \mathcal{R}, \pi_S(\phi_P(a)) \rangle$  and see, whether there exists a  $R_2(a, b) \in \pi_S(\phi_P(a))$  (or a  $R_3(b, a) \in \pi_S(\phi_P(a))$ ), such that  $R_2$  is a subrole of  $R$  (or  $R_3$  is a subrole of  $R^-$ ). Thus, relation checking can be performed locally on one node again.

An extended decision problem is instance retrieval for a concept  $C$ , i.e. we want to find all named individuals  $a$ , such that  $\mathcal{O} \models a : C$ . The idea is that we determine first local solutions on each node, and then use a chosen master node to combine the results. More formally, given nodes  $node_1, \dots, node_n$  let  $\{ap_{i,1}, \dots, ap_{i,m}\}$  denote the ABox partitions associated to node  $i$ . We set  $localresults_i = \{a \mid \exists j. ap_{i,j} \in node_i \wedge \langle \mathcal{T}, \mathcal{R}, \pi_S(\phi_P(a)) \rangle \models a : C\}$ . Then the result of instance retrieval is the union of all the local results, i.e.  $\bigcup_{1 \leq i \leq n} localresults_i$ . Relation retrieval, i.e. finding all pairs of individuals connected by a role  $R$  can be handled in a similar fashion.

Last, we want to look into answering grounded conjunctive queries, without giving a formal definition. We rather want to provide the intuition and leave concrete results for future work. For  $\mathcal{ALCHI}$  grounded conjunctive queries can be answered in the following way:

1. Retrieve the results for all concept query atoms in the query by instance retrieval
2. Retrieve the results for all role query atoms in the query by relation retrieval
3. Combine results from 1) and 2) to answer the grounded conjunctive query

We have provided modular solutions for instance retrieval and relations retrieval above. For the combination of the results, we propose to use a centralized relational database system, such that we have

- one table for each concept query atom (one column corresponding to the individuals which match) and
- one table for each role query atom (two columns corresponding to the pairs of individuals which match) in the conjunctive query.

The idea is that the nodes fill the tables with their local information obtained from local instance and relation retrieval. Then, in the centralized system, the grounded conjunctive query is translated to a SQL query and executed. The result is then a table with individual tuples representing solutions for the conjunctive query. An example is given in Example 2.

**Example 2.** Let  $q = \text{Student}(X) \wedge \text{takesCourse}(X, Y) \wedge \text{GraduateCourse}(Y)$  be a grounded conjunctive query. In the centralized database system we would create the three relations  $\text{Student}(X : TEXT)$ ,  $\text{takesCourse}(X : TEXT, Y : TEXT)$  and  $\text{GraduateCourse}(Y : TEXT)$ . The distributed partitioning systems will fill all three tables with their locally obtained results. After all local results are added, the following SQL-query is used to determine all results for the grounded conjunctive query:

```
SELECT X, Y
FROM Student c1, takesCourse r1, GraduateStudent c2
WHERE
  c1.X=r1.X AND
  c2.Y=r1.Y AND
```

Please note that this approach can be further improved. For instance, if we use a stream based relational database system, then we don't have to wait until all local results are available in the centralized database, but we can evaluate the SQL-query incrementally, and thus, decrease initial query answering latency.

## 8 Conclusions

We have introduced means to reason over  $\mathcal{ALCHI}$ -ontologies, which have large amounts of assertional information. Our updatable partitioning approach allows state-of-the-art description logic reasoner to load only relevant subsets of the ABox to perform sound and complete reasoning. In particular, we have proposed a set of partitioning-preserving update transformations, which can be run on demand. Our techniques can be incorporated into the description logic reasoner RACER[5], to enable more scalable reasoning in the future.

In future work, we will investigate the applicability of our proposal to more expressive description logics, e.g. SHIQ. The extension for transitive roles is straightforward. The incorporation of min/max-cardinality constraints in a naive way can be done as well. However, it has to be investigated, whether the average partition size with these naive extensions is still small enough to be feasible in practice. Furthermore, we intend to perform more evaluation on real-world ontologies to provide detailed timing statistics. Especially the case of boot strapping the assertional part of an ontology needs further investigation.

## References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F.: The Description Logic Handbook. Cambridge University Press, New York (2007)
2. Fokoue, A., Kershenbaum, A., Ma, L., Patel, C., Schonberg, E., Srinivas, K.: Using Abstract Evaluation in ABox Reasoning. In: SSWS 2006, Athens, GA, USA, pp. 61–74 (November 2006)
3. Guo, Y., Heflin, J.: A Scalable Approach for Partitioning OWL Knowledge Bases. In: SSWS 2006, Athens, GA, USA (November 2006)
4. Guo, Y., Pan, Z., Heflin, J.: Lubm: A benchmark for owl knowledge base systems. J. Web Sem. 3(2-3), 158–182 (2005)

5. Haarslev, V., Möller, R.: Description of the racer system and its applications. In: Proceedings International Workshop on Description Logics (DL 2001), Stanford, USA, August 1-3, pp. 131–141 (2001)
6. Halashek-Wiener, C., Parsia, B., Sirin, E.: Description logic reasoning with syntactic updates. In: Meersman, R., Tari, Z. (eds.) OTM 2006. LNCS, vol. 4275, pp. 722–737. Springer, Heidelberg (2006)
7. Motik, B., Oberle, D., Staab, S., Studer, R., Volz, R.: Kaon server architecture. WonderWeb Deliverable D5 (2002), <http://wonderweb.semanticweb.org>
8. Motik, B., Shearer, R., Horrocks, I.: Optimized Reasoning in Description Logics Using Hypertableaux. In: Pfenning, F. (ed.) CADE 2007. LNCS (LNAI), vol. 4603, pp. 67–83. Springer, Heidelberg (2007)
9. Nguyen, A.N.: Distributed storage system for description logic knowledge bases. Technical Report (2009),  
<http://www.sts.tu-harburg.de/wandelt/research/NgocThesis.pdf>
10. Stuckenschmidt, H., Klein, M.: Structure-based partitioning of large concept hierarchies. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 289–303. Springer, Heidelberg (2004)
11. Wandelt, S., Moeller, R.: Island reasoning for alchi ontologies. In: Eschenbach, C., Grninger, M. (eds.) FOIS. Frontiers in Artificial Intelligence and Applications, vol. 183, pp. 164–177. IOS Press, Amsterdam (2008)

# Ontology Co-construction with an Adaptive Multi-Agent System: Principles and Case-Study

Zied Sellami<sup>1</sup>, Valérie Camps<sup>1</sup>, Nathalie Aussenac-Gilles<sup>1</sup>, and Sylvain Rougemaille<sup>2</sup>

<sup>1</sup> IRIT (Institut de Recherche en Informatique de Toulouse)

Toulouse University, 118 Route de Narbonne, F-31062 Toulouse cedex 9, France

{Zied.Sellami,Valerie.Camps,Nathalie.Aussenac-Gilles}@irit.fr

<sup>2</sup> UPETEC (Emergence Technologies for Unsolved Problems)

10 avenue de l'europe, 31520 Ramonville, France

Sylvain.Rougemaille@upetec.fr

**Abstract.** Manual ontology engineering and maintenance is a difficult task that requires significant effort from the ontologist to identify and structure domain knowledge. Automatic ontology learning makes this task easier, especially through the use of text and natural language processing tools. In this paper, we present DYNAMO, a tool based on an Adaptive Multi-Agent System (AMAS), which aims at helping ontologists during ontology building and evolution (co-construction process). DYNAMO is based on terms and lexical relations that have been extracted from text. DYNAMO provides an AMAS based module to support ontology co-construction. The ontologist interacts with the tool by modifying the ontology. Then the AMAS adapts to these changes and proposes new evolutions to improve the ontology. A first experiment of ontology building shows promising results, and helps us to identify key issues in the agent behaviour that should be solved so that the DYNAMO performs better.

**Keywords:** Ontology engineering from text, Multi-agent system, Knowledge acquisition.

## 1 Introduction

One way to provide an efficient search on a document retrieval system is to explicitly state the meaning of document contents. On-going research in this area tries to address the problem by tagging and indexing the contents of documents thanks to an organised knowledge representation called ontology.

Ontologies are often used to represent a specification of domain knowledge by providing a consensual agreement on the semantics of domain concepts, or an agreement on the concepts required for a specific knowledge intensive application. An ontology also defines rich relationships between concepts. It allows members of a community of interest to establish a shared formal vocabulary. In short, ontologies are defined as a formal specification of a shared conceptualisation [1] where formal implies that the ontology should be machine-readable and shared, that is accepted by a human group or community. Further, it is restricted to the concepts and relations that are relevant for a particular task or application.

Typically, ontologies are composed of a hierarchy of concepts the meaning of which is expressed thanks to their relationships and to axioms or rules that may constrain the relations or that define new concepts as formulas. Concepts may be labelled with terms that are their linguistic realisations or linguistic clues of their meaning.

Originally, ontologies were supposed to be stable over time. Nevertheless, ontologies may need to evolve because domain knowledge changes, users' needs may be different or because the ontology could be used in a new context or even reused in a new application [2]. The Ontology maintenance may result difficult especially for large or heavy-weight ontologies.

Ontology engineering is a costly and complex task [3]. In the last ten years, ontology engineering from text has emerged as a promising way to save time and to gain efficiency for building or evolving ontologies [4]. However, texts do not cover all the required information to build a relevant domain model, and human interpretation and validation are required at several stages in this process. So ontology engineering remains a particularly complex task when it comes to the observation of linguistic forms to extract ontological representations from a specific document corpus.

Our motivation is to propose a tool facilitating ontology maintenance and evolution by the ontologist. The principle is to provide a system that automatically proposes solutions to be discussed and evaluated. This system learns from the user's feedback. It can be seen as a virtual ontologist that helps the "real one" to carry out ontology learning and evolution from text. We call this process a co-construction or co-evolution.

In this article, we propose DYNAMO, a Multi-Agent System (MAS) that supports ontology co-construction and evolution. After an overview of the DYNAMO context (section 2), we detail the principles of this MAS in the section 3. In section 4 we exhibit preliminary results obtained when building an ontology with DYNAMO. These results are discussed in the same section. In section 6 we presents the improvements we plan to bring to our work as well as our perspectives.

## 2 Dynamic Ontology Co-construction

### 2.1 Context

The study of ontology evolution is part of the DYNAMO<sup>1</sup> (DYNAMic Ontology for information retrieval) ANR<sup>2</sup> (Agence Nationale de la Recherche) funded research project. DYNAMO addresses the improvement of semantic information retrieval driven by user satisfaction in a dynamic context. One of the project original features is to take into account the potential dynamics of the searched document collection, of the domain knowledge as well as the evolution of users' needs.

The DYNAMO project aims at proposing a method and a set of tools that allow the definition and the maintenance of ontological resources from a set of documents. These resources are used to facilitate information retrieval within the corpus by means of semantic indexing.

---

<sup>1</sup> <http://www.irit.fr/DYNAMO/>

<sup>2</sup> <http://www.agence-nationale-recherche.fr/>

Several project partners propose domain specific document collections. ACTIA<sup>3</sup> provides documents covering the area of automotive diagnosis (automotive components, symptoms, engine failures, etc.) written in French, while ARTAL<sup>4</sup> corpus consists in software bug reports written in English, and the partners in charge of the ARKEOTEK<sup>5</sup> project are concerned by French archaeological scientific research papers structured as a set of rules. Thus, one point of importance in DYNAMO tools, is to handle variety of heterogeneous document collections (either in French or in English) for the co-evolution of ontological resources.

## 2.2 The Ontology Model

In the DYNAMO project, the ontology and its lexical component form what we call a Terminological and Ontological Resource (TOR). Such a resource is represented using the OWL<sup>6</sup>-based TOR model proposed in [5]. This model recently evolved to become a meta-model, where concepts and terms are two meta-classes adapted from owl:Class. In this TOR, model ontological elements (concepts) are related to their linguistic manifestations in documents (terms): a term “denotes” at least one concept. This models forms the core of the DYNAMO project as long as term instances (which represent term occurrences), concept instances and relations between instances are used to represent document annotations.

The problem addressed in this paper is how a multi-agent system can be used to build and update a TOR represented with this meta-model and using documents as information sources.

## 2.3 The Adaptive Multi-Agent System (AMAS) Theory

Because of their local computation and openness, MAS are known to be particularly well fitted to dynamic and complex problems. The design of an ontology from the analysis of a corpus is an obviously complex task (as we discussed in the introduction).

The AMAS Theory [6] emphasises on the cooperation between agents to achieve their intended collective function by the way of self-organisation. Each agent in the system tries to maintain a cooperative state, more precisely, it tries to avoid and repair harmful situations (Non Cooperative Situations). According to the AMAS principles, the cooperative behaviour of agents ensures that, the function realised by the system is always adapted to the problem (functional adequacy). The main idea of the work presented here is to take advantage of the AMAS properties to propose an ontology co-construction system that uses as information sources the documents as well as the interactions with the ontologist.

As a result, the DYNAMO MAS proposes some modifications to the ontologist because it evaluates that the ontology can be improved. The system also benefits from the ontologist's answers to its proposals (basically acceptance or rejection); it also allows to strengthen or weaken the confidence in the position of the involved agents.

---

<sup>3</sup> <http://www.ACTIA.com>

<sup>4</sup> <http://www.ARTAL.fr>

<sup>5</sup> <http://www.ARKEOTEK.org>

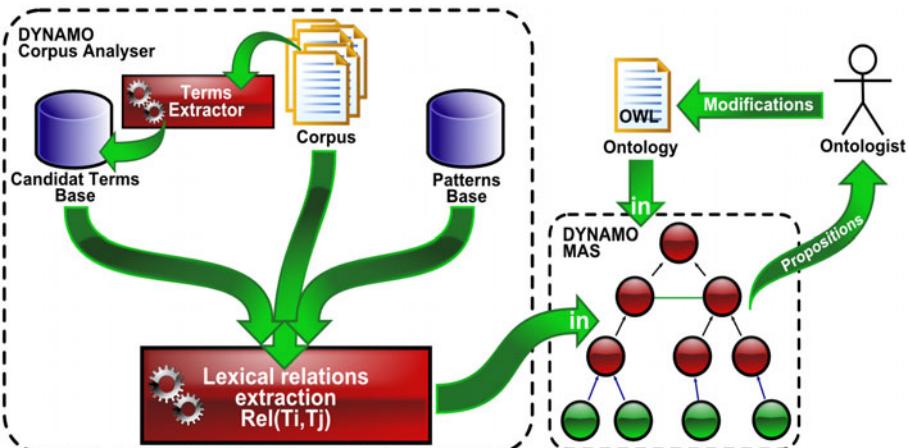
<sup>6</sup> Web Ontology Language <http://www.w3.org/2004/OWL/>

### 3 Principles

DYNAMO is a tool, based on an Adaptive Multi-Agent System (AMAS), enabling the co-construction and the maintenance of an ontology. It takes a textual corpus as input and it outputs an OWL ontology. DYNAMO is a semi-automatic tool because the ontologist has only to validate, refine or modify the organisation of concepts, terms and relations between concepts until it reaches a satisfying state. Figure 1 gives an overview of the DYNAMO system components : *DYNAMO Corpus Analyzer* and *DYNAMO MAS*.

The *DYNAMO Corpus Analyzer* prepares the input of the *DYNAMO MAS*. It contains the *Corpus*, the *Pattern Base*, the *Candidate Term Base* and a set of Natural Language Processing (NLP) tools. Those tools process the lexical relation extraction mechanism the result of which is used to determine potential semantic relations between candidate terms.

The *DYNAMO MAS* is composed of two agent types: *TermAgent* and *ConceptAgent* which are detailed further in Section 3.3. Thanks to the extracted relations these agents try to self-organise in order to find their own location in the TOR hierarchy.



**Fig. 1.** Relations between the *DYNAMO Corpus Analyzer*, the *DYNAMO MAS* and the ontologist

#### 3.1 Syntactic Patterns

Many approaches for ontology learning from text are based on NLP. We can quote two main groups: on the one hand, statistical approaches [7], such as clustering, are interested in finding a semantic interpretation to several kinds of term co-occurrences in corpora; on the other hand, linguistics-based approaches rely on a more or less fine-grained linguistic description of the language used in text to derive an interpretation at the semantic level. Recent ontology learning processes combine both approaches [8].

For instance, lexico-syntactic patterns can be used either for concept or semantic relation extraction, but what they actually identify in text are terms or lexical relations [9]. The extraction process then includes pattern adaptation to the corpus to be parsed, lexical relation extraction on each document, sentence interpretation and finally term

and relation extraction [10]. An extra step would be to define concepts and semantic relations from those items. Systems such as Text-to-Onto [11] or OntoLearn [12] propose a fully automatic run from pattern-matching to ontology learning, while systems like Prométhée [13] and Caméléon [14] support a supervised process where the ontologist may validate or modify the concepts and relations proposed by the analysis tool.

In keeping with results established by V. Malais [15] we have experimentally observed that a statistical processing is not very effective on small corpora with little redundancy, which is the case for the three DYNAMO specific applications. Not only the corpora have a relatively modest size (ACTIA corpus: 46000 words, ARTAL corpus: 13000 words, ARKEOTEK corpus: 106000 words), but each document has a very short length and deals with a specific subject. For all these reasons, we adopted a pattern-based approach to obtain relevant information on terms and their relationships, and then to define concepts and semantic relations from these evidences.

### 3.2 Semantic Relations

In DYNAMO, we are interested in four types of lexical relations:

1. Hyperonymy expresses a generic-specific relation between terms. This may lead to define a class-subclass (*is\_a*) relation between the concepts denoted by these terms.
2. Meronymy means a parthood relation between terms, which may lead to define a *part\_of* semantic relation between concepts, or an *ingredient\_of* relation or domain-specific adaptations of parthood like *has\_members* in biology for instance.
3. Synonymy relates semantically close terms that should denote the same concept.
4. Functional relations: which are any other kind of lexical relations that will lead to a specific set of semantic relations, either general ones like *causes*, *leads\_to*, ... or task specific relations like *has\_fault*, *is\_an\_evidence\_for* or domain specific relations like *has\_skills* in archaeology.

In our system, linguistic manifestations of semantic relations are used by agents as clues for self-organisation. We call them triggers.

### 3.3 Agent Behaviour

The Multi-Agent System is composed of two different types of agent: one representing the terminological part of the TOR (*TermAgent*) and the other, the conceptual part (*ConceptAgent*). Considering an ontology as the MAS organisation, the aim of the DYNAMO MAS system is to reach a stable organisation of terms and concepts according to the semantic relations extracted from the corpus. This is mainly achieved through a cooperative self-organisation process between the agents encapsulating all these elements.

**TermAgent Behaviour.** *TermAgents* represent terms that have been extracted from the corpus. Each extracted term is related to one or many other terms by either syntactic relations or lexical relations, which are linguistic manifestations of semantic relations. These relations have been detected thanks to the lexical relations extraction mechanism (essentially using specific triggers).

The system is initialised by creating one *TermAgent* from each extracted term. The agent behaviour consists then in processing all the extracted relations that connect it

with other terms. Each relation has a confidence degree that is computed from the frequency of occurrences of the corresponding pattern. Using this confidence degree *TermAgents* are related with each other (see 3.2).

Initially the MAS is only composed of *TermAgents* which are linked by these valued relations. Each *TermAgent* takes into account the most important relation (which has the greatest confidence value) according to its type:

1. Each synonymy relation between two *TermAgents* leads to the creation of a *ConceptAgent* linked to the corresponding *TermAgents*.
2. Each hyperonymy relation between two *TermAgents* causes the creation of two *ConceptAgents* linked to the corresponding *TermAgents*. The two new *ConceptAgents* are related by an *is\_a* relation.
3. Each meronymy relation between two *TermAgents* implies the creation of two *ConceptAgents* linked to the corresponding *TermAgents*. The new *ConceptAgents* are associated with a *part\_of* relation.
4. Each functional relation between two *TermAgents* leads to the creation of two *ConceptAgent* linked to their corresponding *TermAgents*. The two new *ConceptAgents* are then related by this relation.

When a *TermAgent* requires the creation of *ConceptAgent*, this *ConceptAgent* establishes a denotation relation towards the *TermAgent*. Additionally, if a *ConceptAgent* with a same identifier already exists, it is not created a second time.

**ConceptAgent Behaviour.** This agent type represents concepts that have been created by *TermAgents*. *ConceptAgents* behaviour behave in order to optimize the ontology. They deal with a set of Non Cooperative Situations (NCS) derived from specific data about the three relation types (hyperonymy, meronymy and functional) as well as their link with terminological data (*TermAgent* denotation). An example of NCS is the election of a preferred concept label. Typically, each *ConceptAgent* has to choose among its related terms the one that is the more representative to become its label. To do so, a *ConceptAgent* selects the denotation relation on which it is the most confident, and proposes to the corresponding *TermAgent* to become its label. However, conflicts may appear in this process. As a single *TermAgent* could denote several *ConceptAgents* it may receive several label requests. This situation is quoted as an NCS (a conflict one); it is detected by a *TermAgent* and should be treated by a *ConceptAgent*. Several solutions could be adopted at this stage:

- If there is only one common *TermAgent* (the label) linked to the concerned *ConceptAgents*, they should be merged.
- If there are several other *TermAgents* linked to the *ConceptAgents*, *ConceptAgents* have to choose another label in their *TermAgents* pool. The *TermAgent* that detects the situation is kept as a label by one of the *ConceptAgents* depending on its similarity to other *TermAgents*.

Between these two cases a mid-term solution has to be found, by considering the number of *TermAgents* linked to each *ConceptAgent*, the similarity between these *TermAgents*, the relation held by the denoted *ConceptAgent*, etc. This is achieved through

cooperation and thanks to the ontologist's actions. The cooperation at the system level is the purpose of the following section.

**Collective Behaviour.** The *DYNAMO MAS* is a real-time system that uses a corpus as input. Each corpus leads to create several hundreds of *TermAgents* and *ConceptAgents*. We need to be sure that the system converges and outputs at least one solution. This convergence is guaranteed by the AMAS theory. In short, because agents are implemented in such a way that they can be considered as cooperative. Their cooperation ensures that the whole set will stabilize after a large set of iterations for information exchange. In fact, the collective process stops when each agent reaches a local equilibrium. This equilibrium occurs when its remaining NCS levels are lower than the NCS levels of its neighbourhood (agents that are related to it). For example, let us consider a given *TermAgent* looking for its relations with other *TermAgents* which are not currently proposed to the ontologist:

- If a neighbour agrees to take their relationship into account, the MAS changes the TOR by adding the new relationship. This modification is proposed later to the ontologist for agreement. If the ontologist disagrees the MAS stores this information to avoid the same request one more time.
- If a related neighbour disagrees, about this modification because of contradictions with other more critical situations in its own neighbourhood, no change occurs.

Furthermore, we considered a minimum confidence threshold in the algorithm, in order to dismiss a large number of non-significant semantic relations extracted from the corpus. The confidence degree of any relation could evolve when new documents are analysed. By this means, relations that were set apart could be later taken into account providing that their confidence degree reaches the threshold. This simple rule also avoids processing relations considered as noise.

The collective solving process of agents is also very efficient for algorithmic reasons:

- the AMAS algorithm assumes a monotonic decreasing of NCS level: typically three or four interactions between agents are sufficient to obtain a local equilibrium;
- when new information arrives in the MAS (coming from new corpus analysis or the ontologist) only the considered agents work. Thus perturbation is very limited inside the MAS.

## 4 Experiments

We experimented our system using the ARTAL corpus defined in the DYNAMO project. This corpus is in English. The objective of the experiment is to evaluate the ontology built up with Agents, and, from this analysis, to improve the *TermAgent* and *ConceptAgent* behaviour.

### 4.1 Settings

The pattern base was fed with some of the triggers for English that are defined by the TerminoWeb project [10]. These triggers are used to extract relations between terms from a corpus (see 3.1). The following list presents some of the chosen triggers:

- for hyperonymy relations: *such as, and other, including, especially*;
- for meronymy relations: *is a part of, elements of, components of*;
- for synonymy relations: *another term for, also called, also known as, synonym*.

The precise numbers of triggers used for each relation type are presented in Table 1. We also used specific triggers for functional relations such as: *when, if, at, on, before, after*.

**Table 1.** Number of triggers for relation extraction

Semantic Relation	Number of Triggers
SYNONYMY	12
HYPERONYMY	21
MERONYMY	23
FUNCTIONAL	16

Due to the small number of triggers, we estimate potential semantic relations between terms by the following rule: if it is possible to find two terms sharing the same contexts in the corpus (i.e. these terms are often used together with similar sets of words), then these terms could be considered as potentially related. The confidence of this relation is calculated using the Jaccard Formula based on the shared context cardinality. For example, the *DYNAMO Corpus Analyser* considered “Web”, “Applet” and “Service” as potential hyperonymy of the concept “Application” because they share the same context.

Finally, we filled the *Candidate terms base* with the terminological part of the ARTAL TOR which contains 692 terms.

## 4.2 Results and Analysis

For the experiment, we have used a strict matching of terms on the corpus, we have not looked for any of their orthographic variations. This option explains the relatively small number of matched relations and facilitates the readability of MAS results. Using *DYNAMO Corpus Analyser*, we have extracted 476 relations between 503 terms. Table 2 gives the number of relations detected using triggers.

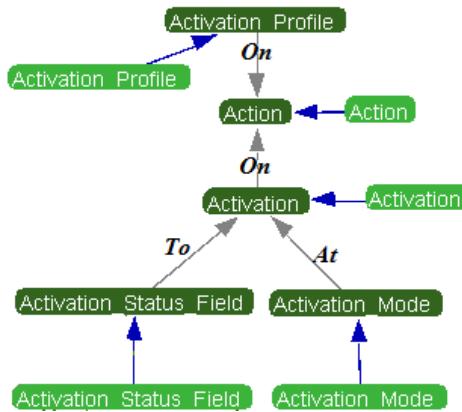
The small amount of synonymy, hyperonymy and meronymy relations is due to the use of generic triggers that are obviously not adequate for this corpus. On the opposite, we used specific triggers to extract functional relations and this is why we get best results. This point highlights the main feature of pattern based approaches which is the strong dependencies of the results on the definition of pattern.

Thanks to the Jaccard formula we have extracted 650 potential semantic relations between terms. The MAS input is formed by 657 candidate terms and 1124 instance of relations.

Firstly, all terms are *agentified* and every agent tries to be related with a *ConceptAgent*. In Figure 2, links without label represent denotation relations between *TermAgents* and *ConceptAgents*. In the ontology model, each term can denote several concepts. Each labelled link represents the functional relation discovered between terms. For example, *Action* and *Activation* concepts are related by a link labelled *On*.

**Table 2.** Number of matched relations in the ARTAL corpus

Semantic Relation	Number of matched relations
SYNONYMY	2
HYPERNONYMY	68
MERONYMY	36
FUNCTIONAL	370

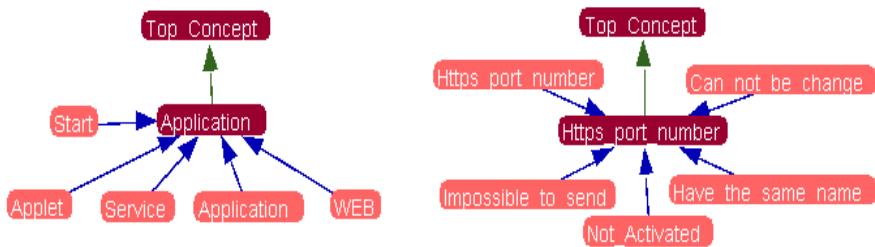
**Fig. 2.** Functional relations established between *ConceptAgents*

In the considered domain (software bug reports) the subgraph in Figure 2 means that an *Action* triggers two kinds of activation. The *Activation* has several modes which can be selected and modifies the status of the associated component (*Activation\_status\_field*). This representation is close to the one proposed by the ontologist, but expresses also some new functional relations.

Figure 3 presents a result obtained with *TermAgent* reasoning only with Jaccard Formula. Five *TermAgents* are related to the “Application” *ConceptAgent* which is related to the top concept. This first graph represents a potential hyperonymy relation between Agents. In the second graph five *TermAgents* (Impossible\_to\_send, Not\_activated, Have\_the\_same\_name, Can\_not\_be\_change, Https\_port\_number) are related to the “Https\_port\_number” *ConceptAgent*. The relation established between this *ConceptAgent* and the first four *TermAgent* represents a potential functional relation (affects). In the considered domain (software bug reports), the relation means that http port can have several bugs such as a non-activation problem.

To improve our results, we need to define more precise patterns by using regular expressions rather than triggers which only detect possible relations between words situated on their both sides. We are also investigating on the use of external information resources (dictionary, generic ontology such as WordNet<sup>7</sup>, other domain related corpus, etc.) to deal with the limitation of corpus based approaches.

<sup>7</sup> <http://wordnet.princeton.edu/>



**Fig. 3.** Example of semantic relation between *TermAgents* and *ConceptAgents* identified with Jaccard Formula

To improve the agent behaviour, we can also combine the Pattern-based approach with a statistical approach by using the Jaccard Formula. For example, a *TermAgent* can improve its own trust in a relation when the same relation is also detected using Jaccard Formula.

As it has been expressed in Section 3.3, we have not completely specified the *ConceptAgent* behaviour, this prototyping phase is part of the ADELFE methodology [16]. The aim of this specific task is to obtain the first draft of the system that allows to highlight more efficiently the NCS. Thanks to these first experiments, we have been able to quote some NCS as, for instance, the label conflict described in Section 3.3.

## 5 Related Works

### 5.1 The Earlier DYNAMO Prototype

The objective of DYNAMO is to facilitate ontology engineering from text thanks to a combination of Natural Language Processing and a cooperative Multi-Agent System. Our research is inspired from DYNAMO first prototype [17] that used a statistical approach to build up a taxonomy from large text corpora. In this prototype, agents implement a distributed clustering algorithm that identifies term clusters. These clusters lead to the definition of concepts as well as their organisation into a hierarchy. Each agent represents a candidate term extracted from the corpus and estimates its similarity with others thanks to statistical features. Several evaluation tests conducted with this DYNAMO first prototype proved its ability to build the kernel of a domain ontology from a textual corpus.

### 5.2 Ontology Engineering from Text in Dynamic Environments

Two on-going major IST European projects, SEKT<sup>8</sup> and NEON<sup>9</sup> aim at similar goals with a more ambitious scope. Both of them are building up toolkits that should give access to a panel of technologies, including several Human Language Technologies among which NLP plays a major role. SEKT and NEON want to advance the state of

<sup>8</sup> <http://www.sekt-project.com/>

<sup>9</sup> <http://www.neon-project.org/web-content/>

the art in using ontologies for large-scale semantic applications in distributed organisations and dynamic environments. Particularly, they aim at improving the capability to handle multiple networked ontologies that exist in a particular context, they are created collaboratively, and might be highly dynamic and constantly evolving. Human Language and Ontology Technologies are combined to produce semi-automatic tools for the creation of ontologies, the population of those ontologies with metadata, and the maintenance and evolution of the ontologies and associated metadata. Although the agents technology is not used at all in these projects, their scope is very similar to the one of DYNAMO.

The ambition of SEKT is to offer this variety of technologies to develop not only ontologies and annotations, but full knowledge management or knowledge intensive applications. Argumentation among ontology authors who locally update an ontology is considered as a key stage of the evolution process of shared ontologies. The NEON project highlights the role of NLP when updating ontologies in dynamic environments together with their related semantic metadata. NEON toolkit offers a tool suite that extends OntoStudio baseline and connects it with GATE. GATE used to propose Protégé as a plug-in for ontology development from text analysis. The new GATE version includes a module that manages its own ontology representation, a plug-in for ontology population and text annotation.

## 6 Conclusions

DYNAMO is a Multi-Agent System allowing the co-construction and evolution of ontologies from text. DYNAMO MAS uses results from lexical relation extraction mechanism to construct ontologies. We shown in Section 4.2 that the system is able to create an ontology draft containing both ontological and terminological elements and enriches several dimensions of the previous prototype:

1. It is able to deal with richer linguistic information as long as agents take into account lexical relations found by matching patterns on text.
2. The result is much richer: DYNAMO builds up a TOR which includes a hierarchy of concepts with their related terms, and labelled semantic relations between concepts. A set of terms denotes each particular concept, which is useful for the document annotation activity.
3. The current DYNAMO system is able to deal either with French or English language text, whereas the first prototype was previously limited to French language.

According to the project schedule we need to improve the software during the next year. To do so we plan to:

- i. introduce the cooperative behaviour of *ConceptAgents* (specification of NCS and their treatment);
- ii. provide an adaptive patterns learning process based on the AMAS theory;
- iii. provide specific interfaces to enable ontologists collaboration;
- iv. apply the *DYNAMO MAS* to all the project domains (archeology, car diagnosis, software bug reports) and languages (French and English).

## References

1. Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition* 5(2), 199–221 (1993)
2. Haase, P., Sure, Y.: D3.1.1.b state-of-the-art on ontology evolution. Technical report, Institute AIFB, University of Karlsruhe (2004)
3. Maedche, A.: Ontology learning for the Semantic Web, vol. 665. Kluwer Academic Publisher, Dordrecht (2002)
4. Buitelaar, P., Cimiano, P., Magnini, B.: *Ontology Learning From Text: Methods, Evaluation and Applications*. IOS Press, Amsterdam (2005)
5. Reymonet, A., Thomas, J., Aussenac-Gilles, N.: Modelling ontological and terminological resources in OWL DL. In: Buitelaar, P., Choi, K.S., Gangemi, A., Huang, C.R. (eds.): *OntoLex07 - From Text to Knowledge: The Lexicon/Ontology Interface Workshop at ISWC 2007 6th International Semantic Web Conference, Busan (South Korea)* (November 11, 2007)
6. Capera, D., Georgé, J.P., Gleizes, M.P., Glize, P.: The AMAS Theory for Complex Problem Solving Based on Self-organizing Cooperative Agents. In: TAPOCS 2003 at WETICE 2003, Linz, Austria, June 9–11. IEEE CS, Los Alamitos (2003)
7. Harris, Z.S.: *Mathematical Structures of Language*. Wiley, New York (1968)
8. Cimiano, P.: *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Springer, Heidelberg (October 2006)
9. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: COLING, pp. 539–545 (1992)
10. Barrière, C., Akakpo, A.: Terminoweb: A software environment for term study in rich contexts. In: Proceedings of International Conference on Terminology, Standardization and Technology Transfer, August 25–26, pp. 103–113. Encyclopedia of China Publishing House, Beijing (2006)
11. Cimiano, P., Völker, J.: Text2onto - a framework for ontology learning and data-driven change discovery. In: Montoyo, A., Muñoz, R., Métais, E. (eds.) NLDB 2005. LNCS, vol. 3513, pp. 227–238. Springer, Heidelberg (2005)
12. Velardi, P., Navigli, R., Cucchiarelli, A., Neri, F.: Evaluation of OntoLearn, a Methodology for Automatic Learning of Domain Ontologies. In: Buitelaar, P., Cimiano, O., Magnini, B. (eds.), IOS Press, Amsterdam (2005)
13. Morin, E.: Using lexico-syntactic patterns to extract semantic relations between terms from technical corpus. In: 5th International Congress on Terminology and Knowledge Engineering (TKE), Innsbruck, Austria, TermNet, pp. 268–278 (1999)
14. Chagnoux, M., Hernandez, N., Aussenac-Gilles, N.: An interactive pattern based approach for extracting non-taxonomic relations from texts. In: Buitelaar, P., Cimiano, P., Palouras, G., Spiliopoulou, M. (eds.) Workshop on Ontology Learning and Population (associated to ECAI 2008) (OLP), Patras (Greece), pp. 1–6 (Juillet 22, 2008)
15. Malaisé, V.: Méthodologie linguistique et terminologique pour la structuration d'ontologies différentielles à partir de corpus textuels. PhD thesis, Paris 7 Denis Diderot University (2005)
16. Bertron, C., Camps, V., Gleizes, M.P., Picard, G.: Engineering Adaptive Multi-Agent Systems: The ADELFE Methodology. In: Henderson-Sellers, B., Giorgini, P. (eds.) *Agent-Oriented Methodologies*, pp. 172–202. Idea Group Pub., NY (Juin 2005) ISBN1-59140-581-5
17. Ottens, K., Hernandez, N., Gleizes, M.P., Aussenac-Gilles, N.: A Multi-Agent System for Dynamic Ontologies. *Journal of Logic and Computation, Special Issue on Ontology Dynamics* 19, 1–28 (2008)

# A Methodology for Knowledge Acquisition in Consumer-Oriented Healthcare

Elena Cardillo, Andrei Tamilin, and Luciano Serafini

FBK-IRST, via Sommarive, 18, 38123 Povo (TN), Italy  
{cardillo,tamilin,serafini}@fbk.eu

**Abstract.** In Consumer-oriented Healthcare Informatics it is still difficult for laypersons to find, understand, and act on health information. This is due to the communication gap between specialized medical terminology used by healthcare professionals and “lay” medical terminology used by healthcare consumers. So there is a need to create consumer-friendly terminologies reflecting the different ways consumers and patients express and think about health topics. An additional need is to map these terminologies with existing clinically-oriented terminologies. This work suggests a methodology to acquire consumer health terminology for creating a Consumer-oriented Medical Vocabulary for Italian that mitigates this gap. This resource could be used in Personal Health Records to improve users’ accessibility to their healthcare data. In order to evaluate this methodology we mapped “lay” terms with standard specialized terminologies to find overlaps. Results showed that our approach provided many “lay” terms that can be considered good synonyms for medical concepts.

**Keywords:** Medical Terminology, Medical Vocabulary Acquisition, Knowledge Acquisition, Consumer Healthcare.

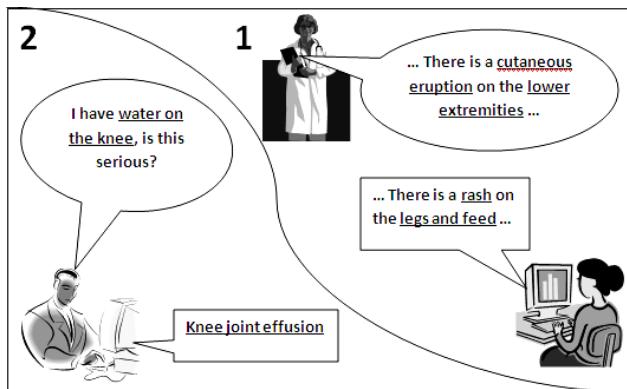
## 1 Introduction

With the advent of the SocialWeb and Healthcare Informatics technologies, we can recognize that a linguistic and semantic discrepancy still exists between specialized medical terminology used by healthcare providers or professionals, and the so called “lay” medical terminology used by patients and healthcare consumers in general. The medical communication gap became more evident when consumers started to play an active role in healthcare information access, becoming more responsible for their personal health care data, exploring health-related information sources on their own, consulting decision-support on the web, and using patient-oriented healthcare systems, which allow them to directly read and interpret clinical notes or test results and to fill in their Personal Health Record (PHRs). As a matter of fact, during this disintermediated interaction consumers can use only their own knowledge, experience and preferences, and this can often generate a wrong inference of the meaning of a term, or the misassociation of a term with its context [15].

Though much effort has been spent on the creation of medical resources (in particular thanks to the improvements in medical Knowledge Integration [10]

facilitated by the use of Semantic Web Technologies [4]) used above all to help physicians in filling in Electronic Health Records (EHRs), facilitating the process of codification of symptoms, diagnoses and diseases (we can see an example in Grabrenweger *et al.* [6]), there is little work based on the use of consumer-oriented medical terminology, and in addition most existing studies have been done only for English. To help healthcare consumers fill this gap, the challenge is to sort out the different ways they communicate within distinct discourse groups and map the common, shared expressions and contexts to the more constrained, specialized language of healthcare professionals.

A consumer-oriented medical terminology can be defined as a “*collection of forms used in health-oriented communication for a particular task or need by a substantial percentage of consumers from a specific discourse group and the relationship of the forms to professional concepts*” [15] (e.g. Nosebleed - Epistaxis; Heart attack – Myocardial Infarction; and other similar relations). It is used most of the time for three possible bridging roles between consumers and health applications or information: a) Information Retrieval, to facilitate automated mapping of consumer-entered queries to technical terms, producing better search results; b) Translation of Medical Records, supplementing medical jargon terms with consumers-understandable names to help patients interpretation; c) Health Care Applications, to help the integration of different medical terminologies and to provide automated mapping of consumer expressions to technical concepts (e.g. querying for the “lay” terms Short of breath and receiving information also for the corresponding technical concepts Dyspnea).



**Fig. 1.** Typical scenario for the use of a CHV

Fig. 1 shows, for instance, how a consumer-oriented medical vocabulary can be useful providing translation functionalities, if integrated in healthcare systems, in two typical scenarios: 1) in the communication from professional to consumer, and 2) in the communication from consumer to professionals. Given this scenario, the present work proposes a hybrid methodology for the acquisition of consumer-oriented medical knowledge and “lay” terminology expressing particular medical concepts,

such as symptoms and diseases, for the consequent creation of a Consumer-oriented Medical Vocabulary for Italian. We are particularly interested in performing analysis of the clinical mapping between this consumer-oriented terminology and the more technical one used in the International Classification of Primary Care (ICPC-2)<sup>1</sup>, to find overlaps between them and to understand how many of these consumer-oriented terms can be used as good synonyms for the ICPC2 concepts. This consumer-oriented resource could be integrated with ICPC2 and other existing lexical and semantic medical resources, and used in healthcare systems, like PHRs, to help consumers during the process of querying and accessing healthcare information, so as to bridge the communication gap. The present work will be structured as follows: In Section 2 is described the State of the Art in the field of medical terminologies, both in clinically and consumer-oriented healthcare; in Section 3 we will present our approach, focusing on the Knowledge Acquisition Process; in Section 4 the Terminology Extraction is presented, followed by the Clinical Review, which is presented in Section 5; finally results and conclusions are presented respectively in Section 6 and Section 7.

## 2 Background

### 2.1 Consumer-Oriented Medical Terminologies

Over the last two decades research on Medical Terminologies has become a popular topic and the standardization efforts have established a number of terminologies and classification systems as well as conversion mappings between them to help medical professionals in managing and codifying their patients' health care data, such as UMLS Metathesaurus<sup>2</sup>, SNOMED International<sup>3</sup>, ICD-10<sup>4</sup> and the already mentioned ICPC-2. They concern, in fact, "*the meaning, expression, and use of concepts in statements in the medical records or other clinical information systems*" [12]. Despite of the wide use of these terminologies, the vocabulary problem continues to plague health professionals and their information systems, but also consumers and in particular laypersons, who are the most damaged by the increased communication gap.

To respond consumer needs to support personal healthcare decision-making, during the last few years, many researchers have labored over the creation of lexical resources that reflect the way consumers/patients express and think about health topics. One of the largest initiatives in this direction is the Consumer Health Vocabulary Initiative<sup>5</sup>, by Q. Zeng and colleagues at Harvard Medical School, resulted in the creation of the Open Access Collaborative Consumer Health Vocabulary (OAC CHV) for English. It includes lay medical terms and synonyms connected to their corresponding technical concepts in the UMLS Metathesaurus. They combined corpus-based text analysis with a human review approach, including

<sup>1</sup> <http://www.globalfamilydoctor.com/wicc/icpcstory.html>

<sup>2</sup> <http://www.nlm.nih.gov/research/umls/>

<sup>3</sup> <http://www.ihtsdo.org/snomed-ct/>

<sup>4</sup> <http://www.who.int/classifications/icd/en/>

<sup>5</sup> <http://www.consumerhealthvocab.org>

the identification of consumer forms for “standard” health-related concepts. Also Soergel *et al.* [14] tried to create such a vocabulary identifying consumer medical terms and expressions used by lay people and health mediators, associating a Mediator Medical Vocabulary with the consumer-oriented one, and mapped them to a Professional Medical Vocabulary. Even in this case the standard terminology used for mapping was UMLS. These and other similar studies examined large numbers of consumer utterances and consistently found that between 20% and 50% of consumer health expressions were not represented by professional health vocabularies. Furthermore, a subset of these unrepresented expressions underwent human review. In most of these cases they performed automatic term extraction from written texts, such as healthcare consumer queries on medical web sites, postings and medical publications. An overview of all these studies can be found in Keselman *et al.* [7].

It is important to stress that there are only few examples of the real application of the most of initiatives. For example, in Kim *et al.* [8] and Zeng *et al.* [16] we find an attempt to face syntactic and semantic issues in the effort to improve PHRs readability, using the CHV to map content in EHRs and PHRs. On the other hand, Rosembloom *et al.* [13] developed a clinical interface terminology, a systematic collection of healthcare-related phrases (terms) to support clinicians’ entries of patient-related information into computer programs such as clinical “note capture” and decision support tools, facilitating display of computer-stored patient information to clinician-users as simple human readable texts. Concerning multilingual consumer oriented health vocabularies, we can only mention the initiative of the European Commission Multilingual Glossary of Popular and Technical Medical Terms<sup>6</sup>, in nine European languages, but it is a limited medical vocabulary for medicinal product package inserts accessible to consumers. In fact, it consists of a list of 1,400 technical terms frequently encountered in inserts, with corresponding consumer terms in all the languages of the EC. Greater overlap between technical and lay terms was observed for the Romance languages and Greek than for the Germanic languages (except English) and some technical terms had no lay equivalent.

## 2.2 Knowledge Acquisition in Healthcare

Knowledge Acquisition process aims at identifying and capturing knowledge assets and terminology to populate a knowledge repository for a specific domain. Central areas of this task are: terminology work, relevant for the special subject field, including terminography; content analysis of documents; extraction of knowledge from various sources. A major part of Knowledge Acquisition is capturing knowledge from experts, a task that is made cost-effective and efficient by using knowledge models and special elicitation techniques. These techniques should be used in different phases of the process, since each of them permits the capture of a specific typology of knowledge and the achievement of specific aims.

The most common techniques are *Interviews*, direct observation of expert performances to extract procedural knowledge, mostly connected to manual skills, such as *Think Aloud Problem Solving*, *Self-report*, and *Shadowing*. Other techniques, such as *Card Sorting*, *Repertory Grid*, and *Twenty Questions*, are useful for

---

<sup>6</sup> <http://users.ugent.be/~rvdstich/eugloss/information.html>

understanding how experts conceptualize knowledge related to their own domain of reference [9].

In a task of Knowledge Acquisition, it is important to identify two main components: knowledge types and modalities, the first components referred to knowledge orientation and domain, and the second ones referred to the representation medium in which knowledge exists. In Knowledge Acquisition for Healthcare domain, according to Adibi [1], many different types of knowledge, which directly contribute to clinical decision-making and care planning, can be identified: Patient, Practitioner, Medical, Resource, Process, Organizational, Relationship, and, finally, Measurement Knowledge. In the present work we will only deal with the Medical Knowledge and the Patient Knowledge. These knowledge types are represented by different knowledge modalities. The most common ones are: Tacit Knowledge of practitioner, Explicit Knowledge, Clinical experiences, Collaborative Problem-solving discussions, and Social Knowledge, etc. In our work we focus on Explicit Knowledge, Clinical Experience and Social Knowledge. In particular the last modality can be viewed in terms of a community of practice and their communication patterns, interest and expertise of individual members.

### 3 Methodology

In this study we focused on a hybrid methodology for the acquisition of consumer-oriented knowledge (lay terms, words, and expressions) used by Italian speakers to identify *symptoms*, *diseases*, and *anatomical concepts*. Three different target groups were considered for the application of our approach: First Aid patients subjected to a Triage Process; a community of Researchers and PhD students with a good level of healthcare literacy; and finally a group of elderly people with a modest background and low level of healthcare literacy. The proposed methodology consists of the following steps:

1. Familiarization with the domain and exploitation of existing common lexical resources (Glossaries, Thesauri, Medical Encyclopedias, etc.);
2. Application of three different Elicitation Techniques to each group:
  - a. Collaborative Wiki-based Acquisition;
  - b. Nurse-assisted Acquisition;
  - c. Interactive Acquisition combining traditional elicitation techniques (Focus Groups, Concept Sorting and Games);
3. Automatic Term Extraction and analysis of acquired knowledge by means of a Text Processing tool;
4. Clinical review of extracted terms and manual mapping to a standard medical terminology (ICPC2), performed by physicians;
5. Evaluation of results in order to find candidate terms to be included in the Consumer-oriented Medical Vocabulary.

### 3.1 Wiki-Based Acquisition

The first method for acquiring consumer-oriented medical knowledge is based on the use of a Semantic Media Wiki system<sup>7</sup>, an easy to use collaborative tool, allowing users to create and link, in a structured and collaborative manner, wiki pages on a certain domain of knowledge. Using our online eHealthWiki<sup>8</sup> system, users created wiki pages for describing symptoms and diseases, using “lay” terminology, specifying in particular the corresponding anatomical categorization, the definition and possible synonyms. The system has been evaluated over a sample of 32 people: researchers, PhD students and administrative staff of our research institute (18 females, 14 males, between 25 and 56 years). Fig. 2 shows an example of wiki page in which users described the symptom “Absence of Voice” (Abbassamento della voce), providing definition in lay terms, anatomical localization, synonyms.

The screenshot shows a Wikipedia-style page for the symptom "Abbassamento della voce". The top navigation bar includes links for "voce", "discussione", "vedi scheda", and "cronologia". The main title is "Abbassamento della voce". Below the title is a large text block starting with "Succede quando non si riesce a far uscire la voce in modo naturale, producendo suoni tenui e rochi, o addirittura si arriva ad essere completamente afoni". There are several sections of text in Italian, including "Può essere localizzato in:", "Può essere indicato anche come:", "Mappa a codice ICPC2:", and "Tipo di corrispondenza tra i termini:". On the left side, there is a sidebar with a logo, a "navigazione" section with links to "Pagina principale", "Guida alla compilazione", "Auto", and "Commenti"; a "dati per categoria" section with "Sintomi" and "Malattie"; and a "sommario" section with "Dati raccolti".

**Fig. 2.** Wiki page example created by users to express the symptom Absence of voice

In one month, we collected 225 wiki pages, 106 for symptoms and 119 for diseases, and a total of 139 synonyms for the inserted terms. It was very interesting to test here also the understanding of the collaborative nature of the Wiki for the specific task, which gave users the possibility to insert not only medical terms by creating wiki pages, but also to update or cancel the inserted information by means of corrections, and above all to modify wiki pages added by other users, in order to reach a convergence on the common sense of medical terminology. In our case, users were reluctant to modify concepts added by others, even in case of evident mistakes in definitions or categorization (only 7 out of 32 provided changes to wiki pages). Some examples of categorization mistakes that had not been modified are “Singhiozzo” (Hiccup), and “Mal di Testa” (Headache), both categorized as Diseases instead of Symptoms. In some cases, when users were in doubt about the right categorization of a concept, they inserted it in both the categories, e.g., “Ustione” (Burning). This test highlighted the fact that users had problems in categorizing medical terms - mainly due to their clinic ambiguity - and also the erroneous use of these terms by them daily.

<sup>7</sup> [http://semantic-mediawiki.org/wiki/Semantic\\_MediaWiki](http://semantic-mediawiki.org/wiki/Semantic_MediaWiki)

<sup>8</sup> <http://ehealthwiki.fbk.eu>

### 3.2 Nurses-Assisted Acquisition

The second technique involved nurses of a First Aid Unit in a Hospital of the Province of Trento<sup>9</sup> as a figure of mediation for the acquisition of terminology about patient symptoms and complaints, helping them to express their problems using the classical subjective examination performed during the Triage Process<sup>10</sup>. This acquisition method involved 10 nurses, around 60 patients per day and a total of 2.000 Triage Records registered in one month. During this period nurses acquired the principal problems expressed by their patients using “lay” terminology and inserted them into the Triage Record together with the corresponding medical concepts usually used for codifying patient data (i.e. the expression “Ho i crampi alla pancia” (I have a stomach ache) inserted together with the corresponding medical concept “Addominalgia” (Abdominal pain).

### 3.3 Focus Group Acquisition

The last method used in our study consisted in merging three different traditional elicitation techniques: Focus Group, Concept Sorting, and Board Games, in order to allow interaction and sharing circumstances to improve the process of acquisition. The target in this case was a community of 32 elderly people in a Seniors Club, aged from 65 to 83 year old. We used groups activities (four groups divided according to a specific body part category, i.e. head and neck, abdomen and back, arms and chest, pelvic area and legs) to acquire lay terms and expressions for symptoms, diseases and anatomical concepts. They were asked to write on little cards all known symptoms and diseases related to the assigned area, comparing their idea with other members of the group to find a common definition for the written terms.

About 160 medical terms were collected, which, at the end of the process, were analyzed together with other groups, creating discussions, exchanging opinions on terms definitions, synonyms, and recording preferences and shared knowledge. In particular, all participants gave preferences for choosing the right body system categorization (digestive, neurological, respiratory, endocrine, etc.) of each of the written concepts. This allowed us not only to extract lay terminology, but also to understand how elderly people define and categorize medical concepts, in order to compare these results with that obtained from the other two mentioned techniques. To give an example of the acquisition process, elderly people in the second group, responsible for the collection of terms related to the body area “abdomen and back”, collected lay terms such as “fuoco di Sant’Antonio” (Shingles) or “sfogo” (Rash) corresponding to the medical term “Herpes Zoster”, describing it as a rash on the lower extremity of the back, due to an allergic reaction to drugs, food or to the contact with plants, and finally they categorized it as a medical concept belonging to the “integument system”.

<sup>9</sup> Medicina d’Urgenza e Pronto Soccorso del Presidio Ospedaliero di Cles (Trento): <http://www.apss.tn.it/Public/ddw.aspx?n=26808>

<sup>10</sup> The Triage activity has the aim to prioritize, by means of a few minutes examinations, patients based on the severity of their condition.

## 4 Term Extraction

Three sets of collected data, including the transcription of the Focus Group activity with elderly persons, were further processed and analyzed, to detect candidate consumer-oriented terms, with Text-2-Knowledge tool (T2K) developed at the Institute of Computational Linguistics of Pisa<sup>11</sup>. This tool allowed us to automatically extract terminology from the data sets and to perform typical text processing techniques (normalization, POS tagging, chunking, etc.), but also to calculate statistics on the extracted data such as term frequency. The computational analysis system adopted by the tool includes a specific plug-in for the analysis of Italian. It provides, as final output, a term-based vocabulary whose added value is represented by the terms' semantic and conceptual information regarding the vocabulary itself. These terms, which can be either single or multi-word terms, are organized in a hierarchical hyponym/hyperonym relation depending on the internal linguistic structure of the terms [2]; that is, by sharing the same lexical head.

In spite of the advantages of the automatic extraction process, allowing for extraction of many compound terms, such a procedure has demonstrated that a good amount of terms, certainly representative of consumer medical terminology, were not automatically extracted, since, due to the quantitative limits of the corpus dimensions, their occurrence was inferior with respect to the predefined threshold value. Consequently, we performed an additional manual extraction to take into account such rare terms, usually mentioned by a single participant.

## 5 Clinical Review

Terms extracted by T2K were further reviewed by two physicians to find mistakes and incongruities in categorization and synonymy. In particular, many mistakes were found by physicians in the first set of terms (Wiki-based), where a wrong categorization was assigned to 25 terms, and where wrong synonyms were expressed for 8 terms. They found similar incongruities in the third set (Elderly people), where wrong categorizations were assigned to 40 terms, e.g. “Giramento di Testa” or “Vertigini” (Vertigo or Dizziness), categorized in the Cardiovascular System instead of the right Neurological one. Concerning the second data set, clinical review was not performed, because it was directly performed firstly by a nurse and then by a physician during the process of Triage.

During the second part of our clinical review physicians were asked to map a term/medical concept pair by using a professional health classification system, the above mentioned International Classification for Primary Care 2nd Edition (ICPC2-E, electronic version) [10], which has received great widespread and preference within the European Union. It addresses fundamental parts of the healthcare process: it is used in particular by general practitioners for encoding symptoms and diagnoses. It has a biaxial structure considering medical concepts related to symptoms, diseases and diagnoses, and medical procedures, according to 17 Problem Areas/Body Systems. In a previous work we encoded ICPC-2-E using the recently developed Web

---

<sup>11</sup> <http://www.ilc.cnr.it>

Ontology Language (OWL) [3] (both for English and Italian), and we also provided the formalization of the existing clinical mapping with the ICD10 classification system, as shown in Cardillo *et al.* [5].

By means of the mapping between “lay” terms and ICPC2 concepts we want to reconstruct the meaning (concept) inherent in the lay usage of a term, and then to agree that consonance between lay and professional terms exists on the basis of this deeper meaning, rather than the lexical form. We identified five different types of relations between consumer terms and ICPC2 medical concepts:

- Exact mapping between the pairs; this occurs when the term used by a lay person can be found in ICPC2 rubrics and both terms correspond to the same concept. For instance, the lay term “Febbre” (Fever) would map to the ICPC2 term “Febbre”, and both will be rooted to the same concept.
- Related mapping; it involves lay synonyms and occurs when the lay term does not exist in the professional vocabulary, but corresponds to a professional term that denotes the same (or closely related) concept. For instance, the lay term “Sangue dal Naso” (Nosebleed) corresponds to “Epistassi” (Epistaxis) in ICPC2.
- Hyponymy relation; this occurs when a lay term can be considered as term of inclusion of an ICPC2 concept. For example, lay term “Abbassamento della Voce” (Absence of Voice) is included in the more general ICPC2 concept “Sintomo o disturbo della voce” (Voice Symptom/Complaint).
- Hyperonymy relation; in this case the lay term is more general than one or more ICPC2 concepts, so it can be considered as its/their hyperonym. For example, the term “Bronchite” (Bronchitis) is broader than “Bronchite Acuta/ Bronchiolite” (Acute Bronchitis/ Bronchiolitis) and “Bronchite Cronica” (Chronic Bronchitis) ICPC2 concepts.
- Not Mapped; those lay terms that cannot be mapped to the professional vocabulary. These can be legitimate health terms, the omission of which reflects real gaps in existing professional vocabularies; or they can represent unique concepts reflecting lay models of health and disease. For example, the lay term “Mal di mare” (Seasickness).

## 6 Results Evaluation

As we have previously mentioned, our methodology of acquisition allowed us to acquire varied consumer-terminology and to perform an interesting terminological and conceptual analysis. Tables 1-4 provide term extraction and mapping evaluation in terms of a statistical analysis. By means of term extraction process, from 225 Wiki pages, we were able to extract a total of 692 medical terms, 375 of which were not considered pertinent to our aim. We performed mapping analysis on 587 terms as summarized in Table 1.

We can observe that most of the exact mappings with ICPC2 are related to anatomical concepts, and which many synonyms in lay terminologies and inclusion terms were found for symptoms. Table 2 shows the results related to the Triage acquisition data.

**Table 1.** Wiki term collection

	<i>Tot. Terms</i>	<i>Exact Map.</i>	<i>Related Map.</i>	<i>Hyponyms</i>	<i>Hyperonyms</i>
Symptoms	306	26	50	40	9
Diseases	140	42	19	38	38
Anatomy	141	105	11	16	4
Other	375	/	/	/	/
Tot.	962				
Not Mapped	186				

**Table 2.** Nurse-assisted term collection

	<i>Tot. Terms</i>	<i>Exact Map.</i>	<i>Related Map.</i>	<i>Not Mapped</i>
Symptoms	508	134	197	177
Diseases	325	86	94	145
Anatomy	275	120	95	60
Other	1281	/	/	/
Tot.	2389			

From 2.000 Triage records, we extracted a total of 2389 terms, but about half of these terms were considered irrelevant for our evaluation, so mapping was provided only for 1108 terms. Contrary to the previous results, here we can highlight the high presence of lay terms used for expressing symptoms with exact mappings to ICPC2, but also many synonyms in lay terminology for ICPC2 symptoms and diseases. This is particularly related to the context chosen for the acquisition, where patients just ask for help about suspected symptoms and complaints. Table 3 shows the results related to the data acquisition from Elderly persons.

**Table 3.** Focus Group/Games with Elderly Person

	<i>Tot. Terms</i>	<i>Exact Map.</i>	<i>Related Map.</i>	<i>Not Mapped</i>
Symptoms	79	35	44	0
Diseases	87	29	54	4
Anatomy	77	51	18	8
Other	78	/	/	/
Tot.	321			

Concerning the last data set, 321 medical terms were extracted by the transcription of the Focus Group/Game activity. Here is interesting to note that all the symptoms extracted had corresponding medical concept in ICPC2 terminology.

Table 4 compares the three data sets together and shows that the most profitable methodology for acquiring consumer-oriented medical terminology was the one assisted by Nurses. But the limit of this method is that it is time-consuming for nurses who have to report all patient “lay” health expressions. While Wiki-based method, even if not exploited for the collaborative characteristic, has demonstrated good qualitative and quantitative results. Concerning the third method we can say that, to be compared with the other two in terms of quantitative results has to be applied more than for 3 hours. On the contrary are interesting the results concerning mapping to ICPC2, because 2/3 of the terms extracted are covered by ICPC2 terminology.

**Table 4.** Results Overview

Sources	Tot. Terms	Tot. Mapped	Not Mapped
eHealthWiki	962	398	186
Nurse-assisted	2389	726	382
Focus Groups	321	231	12
Tot.	3662	1355	580

To conclude our evaluation we have to highlight that comparing the three sets of extracted terms, the overlap is only of 60 relevant consumer medical terms. The total overlap with ICPC2 is about 508 medical concepts on a total of 706 ICPC2 concepts. This means that all the other mapped terms can be considered synonyms or quasi synonyms of the ICPC2 concepts. The large number of not mapped terms and the low overlap between the three sets of extracted terms demonstrate that we extracted a very variegated range of medical terms, many compound terms and expressions, which can be representative of the corresponding technical terms present in standard medical terminologies, and which can be used as candidate for the construction of our Consumer-oriented Medical Vocabulary for Italian.

## 7 Conclusions and Future Work

In this paper we have presented a hybrid methodology for acquiring consumer-oriented medical Knowledge and Terminology for Italian, consisted of lay expressions and terms used to indicate symptoms, diseases and anatomical concepts. We applied three exploratory elicitation techniques to three different samples of people, and we compared results on the basis of a term extraction process, for statistical analysis, and on a clinical mapping procedure, for finding overlaps between extracted lay terms and specialized medical concepts in the ICPC2 terminology. Comparing our approach with that followed by other researchers mentioned in

Section 2, who developed consumer-oriented health vocabularies working only on big written corpora (forum postings and queries to medical websites), using machine learning algorithm and statistical methods (naïve Bayesian classifiers, C-value, etc.) to extract consumer-oriented terminology, we gave more importance to qualitative data, focusing on different methods for acquiring medical lay terminology and knowledge directly from consumers in different scenarios related to General Practice.

This allowed us not only to acquire data but also to try to understand how consumers make good or wrong use of medical terminology, how common expressions daily used in health communication really match to medical concepts used by professionals. In practical terms, our methodology showed encouraging results because it allowed us to acquire many consumer-oriented terms, a low overlap with ICPC2 medical concepts, and a high number of related mappings (most of the time synonyms) to the referent medical terminology. Taking each of these acquisition techniques alone, we have to admit that one of their limits is that they do not allow extraction of lay terminology with a good coverage of the whole domain of pathology and symptomatology. But using a hybrid approach in merging these techniques, and involving a more varied sample of people would improve the results, both from the qualitative and the quantitative point of view. Another limit could be seen in the process of manual mapping performed by physicians. After this pilot study we plan to perform a semi-automatic procedure for mapping lay and specialized terminology, which will be associated to the process of automatic term extraction, and validated by the review of physicians.

To improve the results of the Knowledge Acquisition process and to extract more variegated consumer-oriented terminology, not related to the regional context, we are analyzing written corpora, which include forum postings of an Italian medical website for asking questions to on-line doctors<sup>12</sup>. This would allow extending our sample, covering a wider range of ages, people with different background and consequently different levels of healthcare literacy. This task will be very interesting for comparing results with that came out from the previous elicitation methods, both in quantitative and qualitative terms. Data extracted in this way will be used to validate the acquired terminology, by providing preferences between terms according to frequency and familiarity score.

**Acknowledgements.** We would like to thank Antonio Maini and Maria Taverniti, who provided us with useful support respectively in the process of Knowledge Acquisition and Term Extraction. Finally we would remember that this work is supported by the TreC Project, funded by the Province of Trento.

## References

1. Abidi, S.S.R.: Healthcare knowledge management: The art of the possible. In: Riaño, D. (ed.) K4CARE 2007. LNCS (LNAI), vol. 4924, pp. 1–20. Springer, Heidelberg (2008)
2. Bartolini, R., Lenci, A., Marchi, S., Montemagni, S., Pirrelli, V.: Text-2-knowledge: Acquisizione semi-automatica di ontologie per l'indicizzazione semantica di documenti. Technical Report fot the PEKITA Project, ILC. Pisa, p. 23 (2005)

---

<sup>12</sup> <http://medicitalia.it>

3. Bechhofer, S., Van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F., Stein, A.L.: OWL Web Ontology Language Reference. W3C Recommendation (2004)
4. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *Scientific American* 284(5), 34–43 (2001)
5. Cardillo, E., Eccher, C., Serafini, L., Tamilin, A.: Logical analysis of mappings between medical classification systems. In: Dochev, D., Pistore, M., Traverso, P. (eds.) *AIMSA 2008. LNCS (LNAI)*, vol. 5253, pp. 311–321. Springer, Heidelberg (2008)
6. Grabenweger, J., Duftschmid, G.: Ontologies and their Application in Electronic Health Records. In: *eHealth 2008 Medical Informatics meets eHealth*, Wien, May 29-30 (2008)
7. Keselman, A., Logan, R., Smith, C.A., Leroy, G., Zeng, Q.: Developing Informatics Tools and Strategies for Consumer-centered Health Communication. *Journal of Am. Med. Inf. Assoc.* 14(4), 473–483 (2008)
8. Kim, H., Zeng, Q., Goryachev, S., Keselman, A., Slaughter, L., Smith, C.A.: Text Characteristics of Clinical Reports and Their Implications for the Readability of Personal Health Records. In: *The 12th World Congress on Health (Medical) Informatics, MEDINFO 2007*, pp. 1117–1121. IOS Press, Amsterdam (2007)
9. Milton, N.R.: *Knowledge Acquisition in Practice: A Step-by-step Guide*. Springer, London (2007)
10. Nardon, F.B., Moura, L.A.: Knowledge Sharing and Information Integration in Healthcare using Ontologies and Deductive Databases. In: *MEDINFO 2004*, pp. 62–66. IOS Press, Amsterdam (2004)
11. Okkes, I.M., Jamoullea, M., Lamberts, H., Bentzen, N.: ICPC-2-E: the electronic version of ICPC-2. Differences from the printed version and the consequences. *Family Practice* 17, 101–107 (2000)
12. Rector, A.: Clinical Terminology: Why is it so hard? *Methods of Information in Medicine* 38(4), 239–252 (1999)
13. Rosembloom, T.S., Miller, R.A., Johnson, K.B., Elkin, P.L., Brown, H.S.: Interface Terminologies: Facilitating Direct Entry of Clinical Data into Electronic Health Record Systems. *Journal of Am. Med. Inf. Assoc.* 13(3), 277–287 (2006)
14. Soergel, D., Tse, T., Slaughter, L.: Helping Healthcare Consumers Understand: An “Interpretative Layer” for Finding and Making Sense of Medical Information. In: *The International Medical Informatics Association’s Conference, IMIA 2004*, pp. 931–935 (2004)
15. Zeng, Q., Tse, T.: Exploring and Developing Consumer Health Vocabularies. *J. of Am. Med. Inf. Assoc.* 13, 24–29 (2006)
16. Zeng, Q., Goryachev, S., Keselman, A., Rosendale, D.: Making Text in Electronic Health Records Comprehensible to Consumers: A Prototype Translator. In: *The 31<sup>st</sup> American Medical Informatics Association’s Annual Symposium, AMIA 2007*, pp. 846–850 (2007)

# Combining Statistical and Symbolic Reasoning for Active Scene Categorization

Thomas Reineking<sup>1</sup>, Niclas Schult<sup>1</sup>, and Joana Hois<sup>2</sup>

<sup>1</sup> Cognitive Neuroinformatics, University of Bremen

Enrique-Schmidt-Straße 5, 28359 Bremen, Germany

<sup>2</sup> Research Center on Spatial Cognition SFB/TR 8, University of Bremen

Enrique-Schmidt-Straße 5, 28359 Bremen, Germany

{trking, nschult}@informatik.uni-bremen.de

joana@informatik.uni-bremen.de

**Abstract.** One of the reasons why humans are so successful at interpreting everyday situations is that they are able to combine disparate forms of knowledge. Most artificial systems, by contrast, are restricted to a single representation and hence fail to utilize the complementary nature of multiple sources of information. In this paper, we introduce an information-driven scene categorization system that integrates common sense knowledge provided by a domain ontology with a learned statistical model in order to infer a scene class from recognized objects. We show how the unspecificity of coarse logical constraints and the uncertainty of statistical relations and the object detection process can be modeled using Dempster-Shafer theory and derive the resulting belief update equations. In addition, we define an uncertainty minimization principle for adaptively selecting the most informative object detectors and present classification results for scenes from the LabelMe image database.

## 1 Introduction

Domain ontologies and statistics are fundamental and extensively-used tools for modeling knowledge about the world. On the one hand, domain ontologies provide common-sense knowledge by expressing necessary and general logical relations between entities. These relations reflect the definitions or inherent determinations of the entities involved, or they reflect general rules or laws effective in the domain under consideration. Degrees of belief, on the other hand, account for the fact that perception of the world is intrinsically uncertain. They are usually obtained in an empirical process and thus belong to the realm of statistics. These two forms of knowledge are complementary in nature since ontological models generally abstract from the uncertainty associated with perception while statistical models are restricted to low-order relations which are subject to noise and lack embedding in background knowledge. We therefore argue that the combination of both forms of representations can increase the reasoning robustness for many problems which involve uncertainty and are difficult to model by statistics alone.

A good example of a difficult reasoning problem where both ontologies and statistics can offer valuable information is that of visual scene categorization. Distinguishing between semantic classes of scenes is essential for an autonomous agent in order

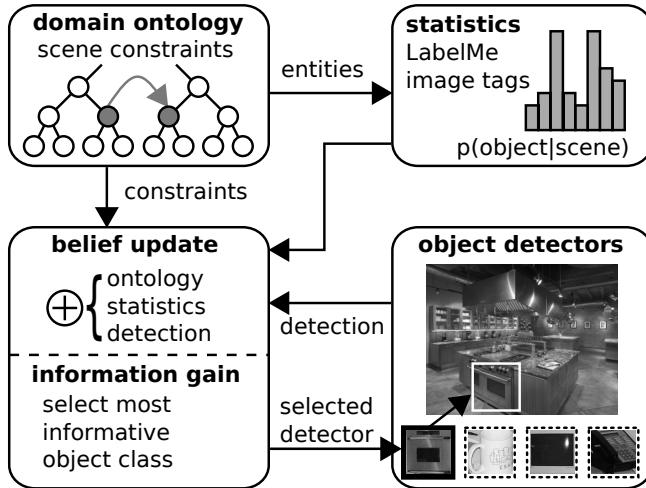
to interact with its environment in a meaningful way. While coarse perceptual classes (e.g., coast, forest) can be recognized from low-level visual features [1], categorizing scenes that mainly differ with respect to activities an agent could perform in them (e.g., mall-shopping, kitchen-cooking) requires an object-centric analysis. The problem thus consists of recognizing objects and combining this information with knowledge about the relations of object and scene classes. These relations can either be of statistical nature (*70% of all street scenes contain cars*) or they can be categorical (*bedrooms contain beds*). The former can be naturally represented by object-scene co-occurrence statistics whereas the latter can be best modeled by domain ontologies.

The problem of assigning a semantic class label to a scene on the basis of sensory information has been discussed in several works recently. In [2], a domain ontology is used for inferring room concepts from sensorimotor features associated with objects. Objects are analyzed via saccadic eye movements which are generated by an information gain maximization process. In contrast to our approach, the system does not distinguish between categorical and statistical knowledge, but rather uses expert knowledge for expressing degrees of belief about relations of room concepts and objects. In [3], semantic scene categorization is used for enriching the spatial representation of a mobile robot. The primary focus here is on learning a strong scene classifier based on range measurements without utilizing knowledge about objects or modeling the uncertainty of the reasoning process. An example of utilizing co-occurrence statistics for modeling context was proposed in [4]. The authors obtained their object co-occurrence model via dense sampling of annotations from the image hosting website *Flickr*. Using this information, a robot was able to predict the locations of objects based on previously observed objects. In the pattern recognition context, [5] introduce an ontology-supported pollen grain classification system. Here, a domain ontology is used to link learned features to visual concepts like color and texture in order to obtain a high-level description of classes.

In this paper, we propose an information-driven architecture that combines a domain ontology with a statistical model which we apply to the problem of scene categorization based on detected objects. An overview of the system architecture is given in the next section. The knowledge representation component that is specified by the domain ontology and the statistical model is described in section 3. Section 4 explains the belief update and the system's uncertainty minimization strategy. Object detection is discussed in section 5 along with results for the classification of scenes from the LabelMe image database [6]. The paper concludes with a discussion of the presented architecture and future work.

## 2 System Architecture

The proposed scene categorization system is composed of four main components: a domain ontology for scenes, a statistical model, a reasoning module for actively updating the scene belief by minimizing uncertainty, and an image processing module consisting of class-specific object detectors (see Fig. 1). Whenever an object detector is invoked, the positive or negative response induces a belief for the presence of the corresponding object class in the current scene depending on the estimated recognition rate of the



**Fig. 1.** Overview of the system architecture showing the four main components of the scene categorization system

detector. This object class belief is then combined with the object-scene knowledge obtained from the domain ontology and the statistical model for updating the scene class belief. Here, the domain ontology defines the vocabulary of object and scene classes as well as occurrence constraints between the two (e.g., *kitchens contain cooking facilities*). The statistical model, on the other hand, provides co-occurrence probabilities of object and scene classes which are learned from annotations available in the LabelMe image database. In order to combine the uncertainty resulting from the object detection and from the statistical model with the set-based propositions from the ontology, we use Dempster-Shafer theory since it allows assigning belief values to arbitrary sets of propositions and thus is capable of representing both uncertainty and logical constraints.

Based on the current scene belief, the system selects the most informative object class for the subsequent detection, i.e., it selects the object class which minimizes the expected scene class uncertainty. The expected uncertainty reduction depends on the discriminatory power of the object class (e.g., ‘stove’ when having to distinguish between kitchens and offices) as well as on the recognition rate of the corresponding detector (if stoves are hard to recognize, their discriminatory power is of little use). Each detector is trained on a large set of sample images using boosting to obtain a strong binary classifier, which is systematically applied to different image regions during the detection phase. In addition, each detector is evaluated on a separate data set for estimating its recognition rate.

Overall, the architecture analyzes a scene in a cycle of bottom-up object detection followed by a belief update based on statistical and logical inference, and top-down uncertainty minimization for selecting the next object class for detection. In order to classify a scene, the system first computes the expected scene uncertainty reduction associated with searching for a specific object class in the scene. This is particularly useful in case the context induces a prior belief so that irrelevant objects are ignored from the

beginning. After selecting the most informative object class, the vision module invokes the corresponding object detector and updates the current scene belief depending on the detection result, the constraints defined by the ontology for this object class, and the co-occurrence probabilities of the statistical model.

### 3 Knowledge Representation

The underlying knowledge representation of the system comprises (i) statistical and (ii) ontological information on the domain which are described in detail in the subsections below. Statistical information results from an empirical process, in our case from analyzing annotations in the LabelMe database. Statistics are generally subject to noise and biases, and they depend on the availability of sufficient sample data. Furthermore, they are restricted to representing low-order relations since the complexity of representation and data acquisition increases exponentially with the number of variables.

While statistical information reflects the probability of relations between objects and scenes, ontological information reflects logically strict constraints and relations between them. A domain ontology for scenes primarily has to formalize the kind of scenes and objects that exist as well as their relationships. In contrast to statistics, it does not rely on a sample set of data, but on expert knowledge and general common-sense knowledge of the domain. It may especially be formalized on the basis of foundational developments in ontological engineering, as outlined below. In essence, (i) statistics contribute a probabilistic correspondence between objects and scenes obtained from a (finite) data set, while (ii) the domain ontology contributes a formalization of entities and relations that exist in the domain, which also provides the vocabulary for the statistics.

#### 3.1 Ontological Model

The domain ontology for visual scene recognition provides the system with information on scenes, objects in scenes, and their relations. Although ontologies can be defined in any logic, we focus here on ontologies as theories formulated in description logic (DL) [7]. DL is supported by the web ontology language OWL DL 2 [8]. Even though ontologies may be formulated in more or less expressive logics, DL ontologies have the following benefits: they are widely used and a common standard for ontology specifications, they provide constructions that are general enough for specifying complex ontologies, and they provide a balance between expressive power and computational complexity in terms of reasoning practicability [9]. Our scene categorization system uses a domain ontology for specific scenes (**SceneOntology**<sup>1</sup>), which is formulated in OWL DL 2. Furthermore, the system uses Pellet [10] for ontological reasoning.

In our scenario, the ontology provides background knowledge about the domain to support scene categorization. Its structure adopts methods from formal ontology developments. In particular, it is a logical refinement [11] of the foundational ontology DOLCE [12]. For practical reasons, we use the OWL version DOLCE-Lite. The domain ontology for scenes conservatively extends DOLCE-Lite, i.e., all assertions made

---

<sup>1</sup> [http://www.ontospace.uni-bremen.de/ontology/domain/  
SceneOntology.owl](http://www.ontospace.uni-bremen.de/ontology/domain/SceneOntology.owl)

in the language of DOLCE-Lite that follow from the scene ontology already follow from DOLCE-Lite. Essentially, this means that the scene ontology completely and independently specifies its vocabulary, i.e., it can be seen as an ontological module [13].

Reusing DOLCE ensures that the domain ontology is based on a well-developed foundational ontology. Its types of classes and relations can be re-used in order to inherit their axiomatizations. Particular ontological classes specified in the scene ontology that are involved in the scene recognition process are `SceneClass`, `SceneEntity`, and `Scene`. Their refinements of DOLCE-Lite are defined as follows:

$$\text{Scene} \sqsubset \text{SceneEntity} \sqsubseteq \text{dolce:physical-object}$$

$$\text{SceneClass} \sqsubseteq \text{dolce:non-physical-object}$$

The class `dolce:physical-object` is a subcategory of `physical-endurant` in DOLCE, which represents those entities that have a physical extent and which are wholly present in time [12]. `Scene` and `SceneEntity` are subclasses of this `dolce:physical-object`. The class `SceneEntity` represents physical entities that occur in spatial scenes and that correspond to segmented objects in scene images. These entities are determined by their intrinsic (inherent) properties. Examples are `Furniture`, `Refrigerator`, `Chair`, `Appliance`, `Tree`, and `Plant`, namely entities that are contained in indoor and outdoor scenes. These scenes are represented by the class `Scene`. The relation `contain` specifies precisely the relation that certain instances of `SceneEntity` are contained in a certain `Scene`. The class `Scene` can be informally described as a collection of contained `SceneEntity`s. In practice, it is related to a specific view on the environment that is perceived by the visual system.

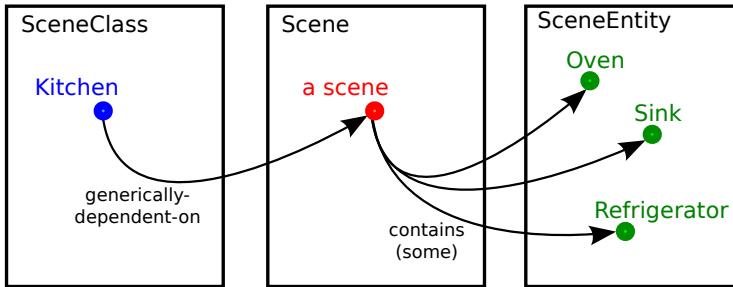
In contrast to `Scene` and `SceneEntity`, `SceneClass` formalizes the type of the scene, i.e., it indicates the category of collections (`Scene`) of entities (`SceneEntity`), illustrated in Fig. 2. Examples are `Kitchen`, `Office`, `ParkingLot`, and `MountainScenery`. `SceneClass` is a subclass of `dolce:non-physical-object`, which is an endurant that has no mass. It constantly depends on a physical endurant, in this case, it depends on the collection of entities that are physically located at a certain `Scene` or that are ‘commonly’ perceivable in this scene. In the scene ontology, `SceneClass` therefore defines the DOLCE-relation `generically-dependent-on` to one `Scene`, which is defined by a conjunction of disjunctions of restrictions on those `SceneEntity` that may occur in the scene. Specific subclasses of `SceneClass` and `SceneEntity` are taken from the LabelMe database.

Hence, for a specific `SceneClass`  $s_i$ , there is a number of subclasses  $x_k$  of `SceneEntity` that necessarily have to occur at the `Scene`  $r_j$  of the `SceneClass`  $s_i$  where  $t_k$  denotes the presence/absence of  $x_k$ . Specific subclasses of `SceneEntity` are taken into account by the following conjunction, with  $K_{s_i}$  indicating the index set of `SceneEntity`  $x_k$ , which constrain  $s_i$  as defined by (2), and  $N$  indicating the total number of subclasses of `SceneEntity`:

$$\xi_{s_i} = \bigwedge_{k \in K_{s_i}} t_k \quad \text{with } K_{s_i} \subseteq \{1, \dots, N\}, t_k \in \{0, 1\}. \quad (1)$$

Each `SceneEntity`  $x_k$  is taken from constraints of the `SceneClass`  $s_i$  as follows:

$$\text{generically-dependent-on}(s_i, (\text{contains}(r_j, x_k))). \quad (2)$$



**Fig. 2.** Instances of **SceneClass** are related to a specific instance of a **Scene** (related to the image), in which instances of **SceneEntity** may occur

The distinction being drawn between **SceneEntity** and **SceneClass** is based on an agent-centered perspective on the domain of possible scenes from the LabelMe database. While instances of **SceneEntity** (e.g., chair, refrigerator, or sink) are on the same level of granularity, instances of **SceneClass** (e.g., kitchen, street corner, or warehouse) are on a broader level of granularity and they particularly depend on a collection of the former. The levels of granularity depend on the agent, i.e., in our case the vision system, that perceives its environment, i.e., a specific scene. The ontological representation of entities differing in granularity aspects is grounded in this agent-based (embodied) vision, as outlined, for instance, in [14]. Note, however, that although an ‘open world’ assumption underlies the ontological representation, the ontology takes into account precisely the objects that are classifiable by the system. Currently, the scene ontology distinguishes between 7 different scene classes and 24 different scene objects.

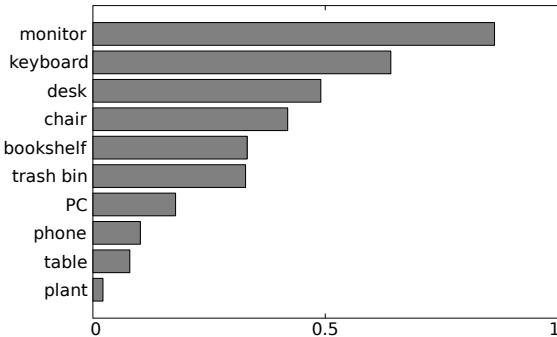
Constraints of specific **SceneClasses**, such as **Kitchen**, are given by the scene ontology on the basis of **SceneEntities**. A sample of such constraints is illustrated in the following example (formulated in Manchester Syntax [15]), which implements the conjunction of disjunction of scene entities that may occur in a scene of a specific scene class:

```

Class: Kitchen
SubClassOf: SceneClass,
    generically-dependent-on only (Scene and (contain some Oven)),
    generically-dependent-on only (Scene and (contain some Sink)),
    generically-dependent-on only (Scene and
        (contain some (Refrigerator or Microwave or CoffeeMachine))),
    ...

```

Queries over such constraints using the reasoner Pellet [16] support the scene recognition process by providing general background knowledge about the domain. Given a request for a specific scene class, the reasoner returns the constraints given by  $\xi_{s_i}$  as formulated in (1).



**Fig. 3.** Conditional probabilities of object occurrences for scene class Office

### 3.2 Statistical Model

The statistical model represents the relation of a scene class  $s_i$  and an object class  $x_k$  by their co-occurrence probability  $p(t_k|s_i)$  where  $t_k$  denotes the presence of  $x_k$ . These conditional probabilities are estimated by computing relative scene-object tag frequencies from the LabelMe database. We restrict our model to these second-order relations since higher-order models exhibit combinatorial complexity, even though this implies ignoring possible statistical dependencies between object classes. After excluding scenes not containing any known object classes, 9601 scenes along with 28701 known objects remain for the statistical analysis. An example of the co-occurrence distribution for the scene class Office is shown in Fig. 3.

## 4 Reasoning

In order to compute the belief about the current scene class, knowledge about scene-object relations from the statistical model and the domain ontology are combined with the object detection responses from the vision module. While the statistical model and object detection can be accurately described by Bayesian probabilities, the constraints defined by the ontology result in propositions about sets of scene classes without any form of belief measure assigned to the single elements within these sets. We therefore use Dempster-Shafer theory [17] throughout the architecture since it generalizes the Bayesian notion of belief to set-based propositions, thus making ignorance explicit and avoiding unjustified equiprobability assumptions. In particular, we use a variant of Dempster-Shafer theory known as the transferable belief model [18], which is based on an open world assumption accounting for the fact that not all scenes can be mapped to the modeled classes.

### 4.1 Belief Update

Let  $\Theta$  be a finite set of mutually exclusive hypotheses. The belief induced over  $\Theta$  by a piece of evidence can be expressed by an (unnormalized) mass function  $m : 2^\Theta \rightarrow [0, 1]$

that assigns values to arbitrary subsets  $A \subseteq \Theta$  (including  $\emptyset$ ) such that  $\sum_A m(A) = 1$ .<sup>2</sup> A mass function can be viewed as an underspecified probability function because there exists a set of compatible probability functions that result from assigning the (normalized) mass value associated with a hypothesis set to its elements. Furthermore, to each mass function  $m$  corresponds a unique plausibility function  $pl$  defined as  $pl(A) = \sum_{A \cap B \neq \emptyset} m(B)$  with  $pl(\emptyset) = 0$ . Like Bayesian probabilities, both mass and plausibility functions can be conditional (denoted by  $\cdot$  due to the lack of normalization).

The aim of the update is to compute the mass distribution  $m(S \cdot d_{k|k \in \hat{K}})$  over the set of scene classes  $\Theta_S$  with  $S \subseteq \Theta_S$  where  $d_k \in \{0, 1\}$  is the binary detector response for object class  $x_k$  and  $\hat{K}$  the set of indices  $k$  corresponding to all object classes  $x_k$  for which a detection was performed up to this point. For deriving this belief, we first apply the generalized Bayesian theorem [19,20]:

$$m(S \cdot d_{k|k \in \hat{K}}) = \prod_{s_i \in S} pl(d_{k|k \in \hat{K}} \cdot s_i) \prod_{s_i \in S^C} (1 - pl(d_{k|k \in \hat{K}} \cdot s_i)). \quad (3)$$

This is the basic update equation of the system. It is important to note that the mass distribution can be computed recursively for each new detection because the plausibilities  $pl(d_{k|k \in \hat{K}} \cdot s_i)$  can be updated recursively and because the plausibilities can be reobtained from the mass distribution using the equality  $pl(A \cdot B) = pl(B \cdot A)$ :

$$pl(d_{k|k \in \hat{K}} \cdot s_i) = \sum_{S \ni s_i} m(S \cdot d_{k|k \in \hat{K}}). \quad (4)$$

The joint detection plausibility  $pl(d_{k|k \in \hat{K}} \cdot s_i)$  can be further simplified by assuming conditional independence between different detection results  $d_k$  and by conditioning [21] on each object class presence  $T_k \subseteq \{0, 1\}$ :

$$pl(d_{k|k \in \hat{K}} \cdot s_i) = \prod_{k \in \hat{K}} \sum_{T_k \subseteq \{0, 1\}} pl(d_k \cdot T_k) m(T_k \cdot s_i). \quad (5)$$

It consists of the object-scene model  $m(T_k \cdot s_i)$  described below and of the single-object detection plausibility  $pl(d_k \cdot T_k)$ . Due to the conditional independence, each new detection simply adds another factor to the joint product and thus allows one to recursively update the joint plausibility from the prior joint plausibility. The single-object detection plausibility can be expressed in terms of the detector rates for true/false positives/negatives using the disjunctive rule of combination [22]:

$$pl(d_k \cdot T_k) = 1 - \prod_{t_k \in T_k} (1 - p(d_k | t_k)). \quad (6)$$

The object-scene model  $m(T_k \cdot s_i)$  in Eq. (5) specifies the object class presence likelihood given a scene class  $s_i$ . As argued in the beginning, this model reflects two sources

---

<sup>2</sup> We use capital letters for denoting sets and minuscule letters for their elements.

of knowledge and can be expressed by the conjunctive combination  $\odot$  of a statistical model  $m^{sta}$  and an ontological model  $m^{ont}$ :

$$\begin{aligned} m(T_k \mid s_i) &= (m^{sta}(\cdot \mid s_i) \odot m^{ont}(\cdot \mid s_i)) (T_k) \\ &= \sum_{T'_k \cap T''_k = T_k} m^{sta}(T'_k \mid s_i) m^{ont}(T''_k \mid s_i). \end{aligned} \quad (7)$$

The statistical model is directly given by the co-occurrence probabilities  $m^{sta}(t_k \mid s_i) = p(t_k \mid s_i)$  described in section 3.2. If these probabilities are not available for a scene class  $s_i$  (e.g., due to a lack of data), the belief is vacuous with  $m^{sta}(\{0, 1\} \mid s_i) = 1$ , expressing a state of total ignorance. The ontological model  $m^{ont}$  is vacuous as well if the corresponding constraint  $\xi_{s_i}$  defined by Eq. (1) does not require the presence of  $x_k$  for scene class  $s_i$ . If, on the other hand, the domain ontology requires the presence of  $x_k$  ( $k \in K_{s_i}$ ) with  $m^{ont}(\{1\} \mid s_i) = 1$ , the absence of  $x_k$  implies the rejection of  $s_i$ . The update for non-atomic constraints (i.e., a scene class requiring the presence of at least one object class from a set of atomic classes) is slightly more complex because the constraint can only be evaluated after all corresponding detectors have been invoked. In this case, the statistical model is ignored since this information was already incorporated and the conditioning in Eq. (5) has to be performed over all the specified object classes.

## 4.2 Information Gain

Aside from passively updating the scene belief in a bottom-up fashion, the system also utilizes a top-down mechanism for selecting object detectors in order to actively reduce uncertainty. If there is little doubt about a scene's class, then it would be wasteful to apply all possible detectors knowing that it would be unlikely to change the scene belief significantly. In order to quantify this change, we need a measure of uncertainty that is applicable to mass functions. We use the local conflict measure  $H(m)$  [23] here since it is a measure of total uncertainty that generalizes the concept of information entropy:

$$H(m) = \sum_{A \subseteq \Theta, A \neq \emptyset} m(A) \log \frac{|A|}{m(A)}. \quad (8)$$

In the context of decision making, the unspecificity represented by mass functions can be ignored by mapping them to classical probabilities [24]. This is achieved by the pig-nistic transformation  $\Gamma$  that maps a mass function  $m$  to a probability distribution  $\Gamma(m)$  by equally distributing masses of non-singleton hypotheses to their corresponding singletons:

$$\Gamma(m)(a) = \sum_{A \subseteq \Theta, a \in A} \frac{m(A)}{|A|(1 - m(\emptyset))}, \forall a \in \Theta. \quad (9)$$

Using these concepts, selecting the most informative object class  $x_{k^*}$  with  $k^* \in \hat{K}^C$  for the subsequent detection is done by minimizing the expected uncertainty of the updated scene belief. For each remaining class  $x_{k'}$  the expected uncertainty is computed by summing over both possible detection outcomes and weighting the resulting uncertainty  $H(m(\cdot \mid d_{k'}, d_{k \mid k \in \hat{K}}))$  of the distribution updated according to Eq. (3) with the

probability  $p(d_{k'}|d_{k|k \in \hat{K}})$  of this event given the previous detections. This probability is computed by conditioning on the scene class  $s_i$  and on the object class presence  $t_{k'}$ . Since we are only interested in the probability here, we apply the pignistic transformation to the prior scene belief and use the plausibility of the object-scene model defined by Eq. (7) instead of the mass value (the latter meaning that the resulting expression is only proportional to the probability):

$$\begin{aligned}
k^* &= \arg \min_{k' \in \hat{K}^C} E_{d_{k'}} \left[ H(m(\cdot|d_{k|k \in \hat{K}})) \right] \\
&= \arg \min_{k' \in \hat{K}^C} \sum_{d_{k'} \in \{0,1\}} H(m(\cdot|d_{k'}, d_{k|k \in \hat{K}})) p(d_{k'}|d_{k|k \in \hat{K}}) \\
&\propto \arg \min_{k' \in \hat{K}^C} \sum_{d_{k'} \in \{0,1\}} H(m(\cdot|d_{k'}, d_{k|k \in \hat{K}})) \\
&\quad \times \sum_{s_i \in \Theta_S} \sum_{t_{k'} \in \{0,1\}} pl(d_{k'}|t_{k'}) pl(t_{k'}|s_i) \Gamma(m(\cdot|d_{k|k \in \hat{K}}))(s_i). \tag{10}
\end{aligned}$$

The extent of the expected uncertainty reduction thus depends on the average detector performance for  $x_{k'}$  on the one hand and on the discriminatory power of  $x_{k'}$  with respect to the current scene belief on the other hand.

## 5 Scene Categorization

In this section, we apply the presented architecture to the problem of categorizing scenes from the LabelMe image database. We first describe the underlying objection detection approach and then continue with presenting quantitative categorization results.

### 5.1 Object Detection

In an earlier work [25], we used the segmentation provided by annotations in the LabelMe database in order to simplify the object detection problem. Here, we instead use a modified version of a detection algorithm made publicly available<sup>3</sup> by A. Torralba. This algorithm does not require any pre-segmentation and is based on learning a strong classifier using GentleBoost [26]. Each detector is trained on 200 positive and 200 negative instances of the corresponding object class. The algorithm first generates a dictionary of filtered patches for each class which perform a weak detection using correlation-based template matching on a  $256 \times 256$  scaled gray-value image generated from the original scene image. These weak detectors are then combined into a single strong detector by applying GentleBoost. After a detector was trained, its error rates are estimated on an additional set of 200 positive scenes that contain the object class and 200 negative scenes that contain no such objects.

---

<sup>3</sup> <http://people.csail.mit.edu/torralba/shortCourseRLOC/boosting/boosting.html>



**Fig. 4.** Example scene showing multiple detected objects

## 5.2 Results

The scene class estimation as described in section 4 consists of two steps: (1) estimating the scene class from present objects and (2) estimating the presence of objects from detector responses. In order to evaluate the former, we first used a set of hypothetical detectors that always give correct responses based on the annotations in the LabelMe database. We defined 7 scene classes and 24 object classes, and we used all available scenes from the database belonging to one of these scene classes if they contained at least one of the defined object classes (3824 scenes total). The overall scene recognition rate in this setup was 94.7%, showing that the combined model accurately describes the domain (correct recognition meant that the actual scene class was assigned the highest plausibility value). The recognition rate varied between scene classes (e.g., offices: 94.3%, living rooms: 84.0%), which can be explained by the fact that some scene classes have more characteristic objects than others. The statistical model alone achieved the same overall recognition rate since the statistical analysis was performed on the same data set used for the recognition task and thus provided a perfect domain model. However, due to the limited number of scenes for some classes (e.g., only 25 bedrooms), this model can not be expected to generalize well. In contrast, the ontological model on its own rarely led to a unique recognition due to the coarseness of the constraints (e.g., *offices contain furniture and electronic equipment*). It did, however, induce a strong prior by restricting the set of possible scene classes to a smaller subset of similar classes (like different outdoor scenes), which always contained the correct class. The iterative analysis of each scene was continued until 97% of the belief mass was committed to a particular class or all available detectors had been invoked. Due to the uncertainty minimization principle, only 7.7 detector invocations were performed on average (of the 24 possible) before reaching this confidence threshold (compared to 14.3 for random selection).

For the actual detectors, we used a more restricted set of object classes and, as a consequence, a more restricted set of scene classes because few object classes provided a sufficient number of training and test samples. One of the reasons for this was that

we used more specific object classes like the front sides of screens, instead of arbitrary perspectives on screens, in order to simplify the object detection, which is still a largely unsolved problem for highly-variable object depictions. Hence, we only used 8 object classes (mean true positive rate: 0.71, mean true negative rate: 0.58) and 2 scene classes (office and street scenes). For each scene class, 200 scenes were randomly sampled from the database under the criterion that they contained at least 1 known object class and had not been used during detector training (Fig. 4 shows one such scene). On this set, the system correctly categorized 91.8% of all scenes. The relative number of detector invocations was higher with 3.7 (of 8 possible; 90% belief threshold) compared to the hypothetical case since the error rates weakened the evidence provided by each detection.

## 6 Discussion

We showed how scene categorization can be performed by combining the complementary information provided by a domain ontology and a statistical model. The consistent propagation of uncertainty from the low-level recognition of objects up to the high-level analysis of scenes enables the system to draw inferences even in case of difficult input images like scenes from the LabelMe image database. We chose Dempster-Shafer theory as a framework for the fusion since it can be used for expressing both set-based implications obtained from the domain ontology as well as probabilistic relations. By using unnormalized mass functions, we made an explicit open world assumption which accounts for the fact that not every scene can be accurately mapped to the set of modeled scene classes. Complementary to bottom-up updating of the scene belief, we presented a top-down reasoning strategy for targeted feature selection based on uncertainty minimization. This selective processing leads to a more efficient analysis of the scene by ignoring irrelevant features, which appears to be an important principle of how humans analyze scenes with saccadic eye movements [27]. Uncertainty minimization can be interpreted as an attention mechanism, and it reflects findings in neuro-psychology showing that object recognition in humans is not simply a feature-driven process, but rather a cognitive process of bottom-up feature extraction and top-down reasoning where recognition is influenced by the context [28].

We argue that domain ontologies and statistics can complement each other since ontologies provide a more general description of the world whereas statistics can offer more detailed but incomplete information which depends on the availability of suited training data. The LabelMe database is a good example for the problem of learning adequate statistical models, because, for many scene classes, it only contains a small number of samples. A more exhaustive domain model in the form of an ontology thus enables the system to reason about classes for which no statistical model might be available at all. There is an interesting analogy between combining different knowledge representations for the reasoning process and the boosting approach used for object detection. The basic idea of boosting is to combine multiple weak classifiers that are only slightly better than random guessing in order to obtain a strong classifier with arbitrary performance. In the same way, one can argue that each source of knowledge contributing more than random guessing improves the overall recognition as long as they represent different aspects of the problem domain.

In the future, we plan to conduct a more comprehensive evaluation of the system's performance by optimizing the underlying object detection approach towards a broader set of classes. Furthermore, we will integrate the scene categorization system into a mobile agent [29]. Not only does this provide a strong prior for the categorization due to the agent's past observations, it also reduces the problem of only partially observing a scene since important objects might be out of sight when only analyzing single images. Finally, we think it would be interesting to see whether the generic approach of reasoning based on ontologies and statistics in a belief-based framework could be applied to other domains beyond scene categorization.

**Acknowledgments.** This work was supported by DFG, SFB/TR8 Spatial Cognition, projects A5-[ActionSpace] and I1-[OntoSpace].

## References

1. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision* 42, 145–175 (2001)
2. Schill, K., Zetsche, C., Hois, J.: A belief-based architecture for scene analysis: From sensorimotor features to knowledge and ontology. *Fuzzy Sets and Systems* 160, 1507–1516 (2009)
3. Martínez Mozos, Ó., Triebel, R., Jensfelt, P., Rottmann, A., Burgard, W.: Supervised semantic labeling of places using information extracted from sensor data. *Robotics and Autonomous Systems* 55, 391–402 (2007)
4. Kollar, T., Roy, N.: Utilizing object-object and object-scene context when planning to find things. In: *International Conference on Robotics and Automation (ICRA)* (2009)
5. Maillot, N.E., Thonnat, M.: Ontology based complex object recognition. *Image and Vision Computing* 26, 102–113 (2008)
6. Russell, B., Torralba, A., Murphy, K., Freeman, W.: LabelMe: a database and web-based tool for image annotation. *International Journal of Computer Vision* 77, 157–173 (2008)
7. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.: *The Description Logic Handbook*. Cambridge University Press, Cambridge (2003)
8. Motik, B., Patel-Schneider, P.F., Grau, B.C.: OWL 2 Web Ontology Language: Direct Semantics. Technical report, W3C (2008), <http://www.w3.org/TR/owl2-semantics/>
9. Horrocks, I., Kutz, O., Sattler, U.: The Even More Irresistible SROIQ. In: *Knowledge Representation and Reasoning (KR)*. AAAI Press, Menlo Park (2006)
10. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. In: *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 5, pp. 51–53 (2007)
11. Kutz, O., Lücke, D., Mossakowski, T.: Heterogeneously Structured Ontologies—Integration, Connection, and Refinement. In: Meyer, T., Orgun, M.A. (eds.) *Advances in Ontologies*, Proc. of the Knowledge Representation Ontology Workshop (KROW 2008), pp. 41–50. ACS (2008)
12. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A.: Ontologies library. Wonder-Web Deliverable D18, ISTC-CNR (2003)
13. Konev, B., Lutz, C., Walther, D., Wolter, F.: Formal properties of modularisation. In: Stuckenschmidt, H., Parent, C., Spaccapietra, S. (eds.) *Modular Ontologies*. LNCS, vol. 5445, pp. 25–66. Springer, Heidelberg (2009)
14. Vernon, D.: Cognitive vision: The case for embodied perception. *Image and Vision Computing* 26, 127–140 (2008)

15. Horridge, M., Patel-Schneider, P.F.: Manchester OWL syntax for OWL 1.1. In: OWL: Experiences and Directions (OWLED 2008), DC, Gaithersberg, Maryland (2008)
16. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. In: Web Semantics: Science, Services and Agents on the World Wide Web, vol. 5, pp. 51–53 (2007)
17. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
18. Smets, P., Kennes, R.: The transferable belief model. *Artificial intelligence* 66, 191–234 (1994)
19. Smets, P.: Belief functions: the disjunctive rule of combination and the generalized Bayesian theorem. *International Journal of Approximate Reasoning* 9, 1–35 (1993)
20. Delmotte, F., Smets, P.: Target identification based on the transferable belief model interpretation of Dempster-Shafer model. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans* 34, 457–471 (2004)
21. Dubois, D., Prade, H.: On the unicity of Dempster's rule of combination. *International Journal of Intelligent Systems* 1, 133–142 (1986)
22. Smets, P.: The nature of the unnormalized beliefs encountered in the transferable belief model. In: Uncertainty in Artificial Intelligence, pp. 292–297 (1992)
23. Pal, N., Bezdek, J., Hemasinha, R.: Uncertainty measures for evidential reasoning II: A new measure of total uncertainty. *International Journal of Approximate Reasoning* 8, 1–16 (1993)
24. Smets, P.: Decision making in the TBM: the necessity of the pignistic transformation. *International Journal of Approximate Reasoning* 38, 133–147 (2005)
25. Reineking, T., Schult, N., Hois, J.: Evidential combination of ontological and statistical information for active scene classification. In: International Conference on Knowledge Engineering and Ontology Development (KEOD) (2009)
26. Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. *Annals of Statistics* 28 (1998), 2000
27. Henderson, J., Hollingworth, A.: High-level scene perception. *Annual Review of Psychology* 50, 243–271 (1999)
28. Schill, K., Umkehrer, E., Beinlich, S., Krieger, G., Zetzsche, C.: Scene analysis with saccadic eye movements: Top-down and bottom-up modeling. *Journal of Electronic Imaging* 10, 152–160 (2001)
29. Zetzsche, C., Wolter, J., Schill, K.: Sensorimotor representation and knowledge-based reasoning for spatial exploration and localisation. *Cognitive Processing* 9, 283–297 (2008)

# A Semi-automatic System for Knowledge Base Population

Jade Goldstein-Stewart<sup>1</sup> and Ransom K. Winder<sup>2</sup>

<sup>1</sup> U.S. Department of Defense, Washington, U.S.A.

jadeg@acm.org

<sup>2</sup> The MITRE Corporation, Annapolis Junction, MD, U.S.A.

rwindr@mitre.org

**Abstract.** The typical method for transferring key information from unstructured text to knowledge bases is laborious manual entry, but automated information extraction is still at unacceptable accuracies to replace it. A viable alternative is a user interface that allows correction and validation of assertions proposed by the automated extractor for entry into the knowledge base. In this paper, we discuss our system for semi-automatic database population and how issues arising in content extraction and knowledge base population are addressed. The major contributions are detailing challenges in building a semi-automated tool, classifying expected extraction errors, identifying the gaps in current extraction technology with regard to databasing, and designing and developing the FEEDE system that supports human correction of automated content extractors in order to speed up data entry into knowledge bases.

**Keywords:** Information extraction, Human-computer interaction, Content analysis, Graphical user interface, Knowledge base population.

## 1 Introduction

With the rapid growth of digital documents, it is necessary to be able to extract identified essential information at a particular time and create knowledge bases to allow for retrieval and reasoning about the information. Unfortunately, database entry is time consuming. If automatic processes could extract relevant information, such methods could automatically populate a “knowledge” base (KB) based on document information. For such knowledge bases to be useful, the end user must trust the information provided, i.e., it must have a high enough degree of accuracy and/or provide a means to correct and validate the information.

In the past two decades research has been dedicated to the automatic extraction of text entities, relations, and events. While the best precision scores for entity extraction are in the 90s [1], precision for relations is typically less than 40%, and events have an even lower precision. Through the Automatic Content Extraction (ACE) program, an ontology has been developed to characterize the types of extraction, and annotation guidelines have been developed to cover ambiguous cases. For example, in the sentence, “the family went to McDonald’s” is McDonald’s a facility, an organization, or both? Is the definition of a facility a place that is locatable on a map?

In ACE, entities [2] can be people, organizations, locations, facilities, geographical/social/political entities, vehicles, or weapons, and their mentions are the textual references to an entity within a document. These can be a name (“Ben Smith”), a representative common noun or noun phrase called a nominal (“the tall man”), or a pronoun (“he”). Although good scores have been achieved in entity tagging, there is cause to doubt the extensibility of systems trained for this task [3]. Also, because an entity can be referred to multiple times, an entity potentially has many mentions, and mentions of the same entity are said to be coreferenced. The best extractor scores for coreferencing entity mentions are in the range of 60-80% [4]. Since relations or events can involve referents of multiple entities, the likelihood of accurately extracting all arguments of a relation or event is low.

Using ACE terminology, a relation [5] is defined as an ordered pair of entities with an asserted relationship of a specific interesting type. So a relation can be thought of as a four tuple: <entity1, relation, entity2, time>. For example, “Scott was a member of ACM for four years” contains a relation where the first entity is “Scott,” the second entity is “ACM,” and the time is a duration of “four years.” This relation has a type of “Organization Affiliation” and has a subtype of “Membership.”

An event [6] is defined as a specific occurrence involving participants and a “trigger” word that best represents the event (e.g., “attacked” in “The rebels attacked the convoy yesterday”). Despite this broad definition, ACE limits its events to a set of types and subtypes that are most interesting. For example, “Jen flew from Boston to Paris” contains a “travel” event, defined as an event that captures movement of people, that is, a change of location of one or more people. The captured arguments of the event would be the travelling entity “Jen,” the origin “Boston,” and the destination “Paris.” Like relations, events can have associated time values [7].

In an examination of a leading rule-based commercial extractor on 230 annotated internal documents, it was able to identify the “ORG-AFF/Membership” relation with a precision of 47% (meaning that 47% of the times it identified this relation, the relation existed in the data). The recall was also 47% meaning that 53% of the membership relations in the data were missed by the system. For those relations that were identified, the first entity, the person, was identified with 71% precision, meaning that 29% of the items that the system returned were incorrect. For the second entity, the organization, the precision was 85%. After the company improved the results, the new relation identification improved to 70% while it remained the same for the two entity arguments. A member of this company suggested that this score was considered “very good” for relations and was unsure that much more improvement could be obtained.

Unfortunately, relations and events are often the key assertions that one needs in a knowledge base in order to identify information about people and/or organizations. Due to the high error rate in extraction technology, rather than introducing errors into the knowledge base, a preferred solution might be semi-automatic population of a knowledge base, involving the presentation of extracted information to users who can validate the information, including accepting, rejecting, correcting, or modifying it before uploading it to the knowledge base. This interface must be designed in a manner that supports the users’ workflow when doing this task. Ideally, the interface would speed up significantly the time to enter data in the knowledge base manually.

Since extractor recall tends to be less than 60%, besides correcting precision errors that the extractor makes, the interface must have the ability for users to add information missed by the extractor (recall errors).

In this paper, we describe the challenges faced in this task and define the design for our system, FEEDE – Fix Extractor Errors before Database Entry. We also discuss the required elements as defined by our end users, the interface’s design, and an examination of the extractors used to populate it with initial content to be authenticated. Given the daunting task of manually entering all important information in a knowledge base from unstructured text, we believe this effort is important to save users time, both a valuable commodity in this information age as well as being enterprise cost saving.

To our knowledge, this is the first research effort on developing an interface using content extraction from unstructured text for populating knowledge bases. It has only been in recent years [8] that the automatic extraction community has started to focus on text extraction for the purpose of populating databases, a field that is to our knowledge not well explored. Past efforts on related topics include an interface effort for structured data (metadata) [9] and specific extraction of biomolecular interaction information from journal articles that also involved human review for catching errors in the automated extraction [10]. Furthermore, since content extraction efforts have not been focused on the database issue, they are missing certain items that are important for such endeavors. A recent survey of extraction elements important to our users revealed that only 25 out of the 47 requested (53%) were in the ACE guidelines.

## 2 Content Extraction for Databasing Issues

ACE provides specifications for tagging and characterizing entities, relations and events in text, as well as some other features. For entities, the key attributes are type and subtype. Mention categories are also important attributes, determining the specificity of the entities, such as a pronoun referent to an entity name. Relations and events also feature types and subtypes as well as arguments—two for relations, where the order matters, and potentially many different arguments for events where the allowed set depends on the event type. Although quite extensive, the ACE guidelines [2,5,6] and temporal annotation guidelines TimeX2 [11] were not designed for databasing and are missing key items necessary for this purpose. Omissions include:

1. Insufficient data elements from current available extractors to cover what our users want. Therefore new extractors must be developed, which requires requirements gathering, definitions and guidelines. Some have been developed for the highest priority items.
2. In the ACE guidelines, values/traits or contact information for people cannot be associated with people (e.g., “John is 6 ft tall”).
3. Nested events do not exist. This is particularly an issue with source attribution. Items attributed to people or a new source are not linked. An example is “According to John, Jane travelled to France.” The fact is not necessarily true, but John states it. Since no such extractor existed, we developed one which linked assertions to people.

4. No mechanism in ACE covers group participation. For example, “Anne and John attended Stanford. In spring 2005, the two decided to travel to Europe.” ACE does not contain a way to reference “two” to Anne and John, although this is a frequent language pattern.
5. ACE lacks a meaningful primary entity name (PEN) for entities. We define an entity’s PEN to be its longest named mention in the document (nominals, titles and pronouns are excluded).
6. ACE lacks descriptors, that is nominals that can define the entity in the context. These important descriptions include titles (e.g., professor), careers (e.g., lawyer), and important roles (e.g., witness). This allows for a distinction between terms that are more of interest than others. We care more that an individual is “prime minister” than about the description of “a man in a green hat.” A simple initial solution for distinguishing these is to have an exhaustive gazetteer of all words in each category that are considered descriptors.
7. ACE lacks sufficient time normalization. Databases can allow one to visualize items linked with temporal information and reason over temporal items, if entries have time stamps. The only available temporal normalizer was TIMEXTAG [11], which did not have sufficient coverage for our purposes. To develop the temporal normalizer, a group of 5 potential users developed grounding rules for key temporal expressions. Users independently mapped all items and then met to come to consensus when there was disagreement. An ambiguous example is “late January,” which maps to 21st-31st January. We hope to make this temporal normalizer available to the public soon.
8. While time tagging guidelines include a methodology for sets, they still need to capture the number of members in the set and how often something occurs. For example, a tag for “the past three winters” has no way of representing “three” and a tag for “twice a month” has no way of representing “twice.” The knowledge base and database need a way to support this information.

This list indicates that much research and development is required before extraction is at a sufficient level for populating knowledge bases.

**Table 1.** Results using value-scorer for RB, ST, and ST2 extractors on newswire data

	ST	RB	ST2
Entities	72.2	72.8	73.1
Entity Mentions	84.6	84.0	84.8
Relations	26.2	24.7	27.3
Events	17.8	N/A	N/A

**Table 2.** Results for entities and relations for all three systems and events for ST, where P is precision and R is recall

	ST Ent.	RB Ent.	ST2 Ent.	ST Rel.	RB Rel.	ST2 Rel.	ST Evt.
Unweighted P	49.8	51.9	51.3	34.8	32.8	39.9	2.2
Unweighted R	53.5	58.5	57.1	22.1	24.1	26.3	1.9
Unweighted F1	51.6	55.0	54.1	27.0	27.8	31.7	2.0

Besides the missing key components, as mentioned the accuracy of content extraction is too low for automatic population of databases and perhaps at levels that could frustrate users. The ACE 2005 value results (an official ACE measure) for newswire documents are presented in Table 1 for three participating systems, two statistical (ST and ST2) and one rule-based (RB). Scores for entities, relations, and events are presented in Table 2. We present ACE 2005 since it had more participation on the relations task than ACE 2007, and ACE 2008 did not evaluate events.

Analysis indicates only slight performance increases for systems in 2007 and 2008. These results were computed using the ACE 2005 evaluation script [12] on each extractor's documents compared to reference documents tagged by humans. In the script's unweighted scores, a relation or event is considered to be located in a system-tagged document if there are no missing reference arguments and no attribute errors. Precision equals the number of these mappable pairs over the total number the extractor found. Recall equals the number of these mappable pairs over the total number in the reference text. These are combined to produce an F1 score.

Note that any error present in the data would need to be fixed by a user that chose to utilize that piece of information in the knowledge base. Thus the trade-off between precision and recall has great significance for the task, as a higher precision would imply less user correction of errors in extracted data, but also requires more manual entry of missing assertions, while a higher recall implies the reverse.

Given the low precision and recall as shown by the ACE 2005 results for relations, we believe it is essential to have an effective user interface to allow users to correct the information extracted incorrectly from the documents (precision errors) as well as to enter missed information (recall errors). Though the results available for events are not as comprehensive, these scores are even lower than those for relations.

In terms of the interface, we define effective as (1) intuitive, (2) easy to use, (3) minimizing mouse clicks, (4) following the workflow, (5) faster than manual entry, (6) tailored to user requirements and preferences, and (7) assisting and guiding the user in completing the task of creating entries for the knowledge base.

Since there are so many potential extraction errors and the knowledge base requires vetted information, the user must validate all information before it is uploaded. Table 3 displays a list of common problems encountered in extraction. Solutions to these issues either require action to be performed by the interface in pre-processing before the information is presented to the end user or in the interface itself, essentially assisting the end user in making corrections based on observations of the evidence.

Something else important to the interface that is absent from ACE is confidence levels in the information. We use an extractor confidence, if provided, as a factor into the confidence score presented to the user. Pronominal references lower the confidence since their accuracy is only approximately 80%. We also use evaluation knowledge about relations and events as a component in the final confidence score. Other factors that could be included might be based on the contextual source; this includes weak attribution, conditional terms, hypothetical statements, or future tense.

Confidence levels indicate to the user whether this assertion has a good chance of being correct, i.e., it helps to focus items they might choose to validate. It is important to note that as we add more information that essentially second-guesses what the extractor has produced, it becomes necessary to distinguish between what is believed and what is not, and confidence plays a role here too.

**Table 3.** Common extraction errors and how the interface handles them

Error Type	Error	Solution Type	Solution
Entity Label	Type/Subtype Error	Preprocess	Perform string matches for the PEN (or any named mention) of entities. If one matches with an entity of interest of different type/subtype, present to User.
Entity Error	Misspelled by Doc Author	Interface	User corrects by typing in the interface. This applies to punctuation and capitalization errors too.
Entity Error	Extent Error	Preprocess	Compare entity of interest (EOI) PENs to text of other entities. If there is significant crossover, offer the EOI as possibility to User for assertions involving such entities.
Entity Error	Extent Error	Interface	User can add missing information or delete extraneous information.
Relation/Event	Argument Error	Preprocess	All proximate entities (mentions within X words) to relation/events should also be offered as alternatives for the actual relation/event.
Relation/Event	Argument Error	Interface	User examines this relation/event and evidence to recognize an incorrect argument and must either ignore, modify or add a new relation/event. Modifying includes selecting a new argument from a drop-down menu list with possible entities for that argument.
Relation/Event	Type/Subtype Error	Interface	User observes this error in the evidence and must either ignore, modify or add new relation/event. Relation/event types can be selected by drop-down menu.
Relation/Event	Spurious Relation/Event	Interface	User observes this relation is spurious in the evidence and can ignore (hide) the relation/event.
Relation/Event	Missing Relation/Event	Interface	If User can recognize this error by viewing the evidence, new relation/event can be added in the interface. Interface supports this with menus listing allowed relations/events.
Coreference	Spurious Coreferences	Interface	User must recognize in evidence that entity mention and primary entity name are different and must either ignore, modify or add new relation/event.
Coreference	Missing Coreferences	Preprocess	All proximate entities (mentions within X words) to relations/events should be offered as possibilities (with low confidence) for the actual relation/event arguments.
Coreference	Split Entities	Interface	When validating, User can assign the same KB id to the two entity chains and the data will be merged in the KB.

### 3 Error Analysis

Because relations and events are the most commonly desired pieces of information to be gleaned from a document, we provide examples of the types of errors observed involving relations and events, drawing from results achieved on the 2005 ACE evaluation. Since the most interesting attributes are type and subtype, in this section we only record attribute errors in these. This change would increase the results in Table 2 by less than 13%. These results are still low and indicate that there are many items that need to be corrected for total accuracy. Here we further examine a leading

statistical extractor (ST) and a leading rule-based extractor (RB). Event results were only available for ST.

In the tables below, we examine the frequency of specific error types independently. Consider the sentence “British Prime Minister Tony Blair left Hong Kong.” This contains a relation of type/subtype “PHYS/Located.” The extent of the first argument is “British Prime Minister Tony Blair,” while the head of the first argument is “Tony Blair.” The extent and head of the second argument are both “Hong Kong.” It is the head—a more specific piece of text—that determines whether two mentions in separately tagged documents map to one another. There is potential for error with any of these elements. Table 4 shows the cases where relations (and events) are tagged with the wrong type or subtype, while Table 5 shows span errors for the full arguments of relations and their heads.

**Table 4.** Relation and event type/subtype error rates observed for newswire documents if other requirements for finding relation or event are filled

	Relations Tagged w/ Incorrect Type/Subtype	Events Tagged w/ Incorrect Type/Subtype
ST	16.6%	11.4%
RB	18.5%	N/A

**Table 5.** Argument span error rates observed across relation mentions. Unmarked results exclude cases of mismatched relation type/subtype, while results marked with an \* ignore the relation type/subtype and just evaluate the head and extent spans.

	Arg1 Head	Arg2 Head	Arg1 Extent	Arg2 Extent	Arg1 Head*	Arg2 Head*	Arg1 Extent*	Arg2 Extent*
ST	2.5%	4.6%	26.9%	13.1%	3.4%	3.8%	25.3%	12.2%
RB	4.9%	8.0%	22.2%	14.8%	5.4%	7.8%	22.9%	16.1%

Considering cases where relations are potentially identified but are tagged with an incorrect type or subtype, the extractors comparably misidentify the types of relations in the reference corpus at rates of 16.6% and 18.5% for ST and RB, respectively.

Turning to the relation’s arguments, their mentions can have errors in the span of their extent (text tagged as being the full argument entity) and head (the key text that defines the argument entity). For both extractors the error rates for extent spans are higher for the first argument than for the second. Head span errors are lower for both extractors, but because relations that do not have overlapping heads will be classified as spurious, it is not surprising that the feature which is the criterion for mapping relations between reference and system-tagged documents has a low error rate when examined.

As Table 6 shows, ST misses 66.9% of these specific relation mentions and RB misses 72.8%. If the relations themselves are considered, as opposed to their specific mentions, then 74.0% of relations are missed by ST and 70.7% of relations are missed by RB. When considering relations as opposed to relation mentions, some of this error is propagated from errors in entity coreferencing. If perfect entity coreferencing is assumed, then the number of missing relations drops to 60.6% for ST and 59.3% for

RB, which is still a high number in both cases. These numbers are still quite high if we permit relations to be recognized that are unmatched with a reference relation and have the appropriate arguments but a different type/subtype, 52.1% for ST and 50.1% for RB. While the advantage in terms of recall belongs to RB, the extraction results on relations are highly errorful, and even when accounting for errors, the precision and recall are quite low, indicating that human validation is necessary for relations. Examining the spurious relations, these make up a significant portion of the returned results. Even presuming perfect entities and ignoring tag mismatches, more than a fourth of returned relations are spurious for either ST or RB.

**Table 6.** Rates of missing and spurious relations. STA/RBa results consider cases of mismatched type/subtype missing or spurious, while STb/RBb results ignore type/subtype.

	Missing Rel. w/ Perfect Coref.	Missing Rel.	Missing Rel. Mentions	Spurious Rel. w/ Perfect Coref.	Spurious Rel.	Spurious Rel. Mentions
STA	60.6%	74.0%	66.9%	40.1%	59.1%	44.8%
RBa	59.3%	70.7%	72.8%	52.9%	60.2%	65.5%
STb	52.1%	68.9%	60.2%	27.2%	51.0%	33.6%
RBb	50.1%	64.1%	65.6%	42.2%	51.1%	56.3%

**Table 7.** Rates of missing and spurious events. The STA results consider cases of mismatched type/subtype missing or spurious, while the STb results ignore type/subtype.

	Missing Evt. w/ Perfect Coref.	Missing Evt.	Missing Evt. Mentions	Spurious Evt. w/ Perfect Coref.	Spurious Evt.	Spurious Evt. Mentions
STA	84.8%	87.6%	87.8%	82.4%	85.7%	84.0%
STb	82.9%	86.0%	87.6%	80.1%	83.8%	83.7%

**Table 8.** Error rates (in %) for spurious and missing arguments for all ST events where STA results consider cases of mismatched type/subtype missing or spurious, while STb results ignore type/subtype. Arguments assigned the wrong role are considered found in the results marked with an \*. Numbers in the parentheses include arguments of missing or spurious events in their counts.

	Missing Args	Missing Args*	Spurious Args	Spurious Args*
STA	54.6 (76.2)	51.0 (74.4)	30.1 (52.3)	24.7 (48.6)
STb	62.4 (74.1)	55.2 (69.1)	41.7 (48.1)	30.4 (38.1)

Table 7 displays the results for missing and spurious events. These occur at very high rates, with 87.8% of specific event mentions missed and 87.6% of events missed. Even when perfect entity coreferencing is assumed, the percent of events missed only drops to 84.8%. As for spurious event mentions, these make up 84.0% of tagged event mentions and 85.7% of tagged events. Once again assuming no errors with entity coreferencing only drops the percentage of spurious events to 82.4%. Table 4 reveals that only 11.4% of events are potentially tagged with the wrong type/subtype. If this restriction is ignored, scores for missing and spurious events improve only marginally.

These numbers are so high mainly due to the difficulty in capturing all reference arguments, a requirement for finding events. Note that events are chiefly defined by their arguments. Examining event arguments more closely, we discover that the error rate for missing arguments is about 54.6%. This error rate increases dramatically though if the totals are allowed to include the missing arguments of completely missing events, rising into the 70s. With regard to spurious arguments, they make up 30.1% of the arguments identified. This number can rise into the 50s if the arguments of completely spurious events are included in the totals. These numbers are presented in Table 8. While results improve when restrictions on event type/subtype and argument role are slackened, they still remain significantly high.

Apart from type, subtype, and arguments, events in ACE are defined by attributes ignored in these results such as tense, genericity, modality, and, perhaps most importantly, polarity. The last of these tells whether or not the event is positive or negative, which means essentially whether or not the event happened. Excluding tense, the scores for these are high, but this is due to the tagger consistently tagging all events with a particular value (its most common). Therefore, the scores for these values are meaningless, which is particularly significant for polarity, as it essentially changes the entire meaning of an event. Considering this as well as the low recall and precision of both events and their constituent parts, it is clear that these also cannot be accurately mediated by automatic means alone but require human validation and correction.

## 4 System, Database and Interface

An analysis of the issues involved in bringing together these different approaches gave rise to a list of challenges that must be dealt with in order to achieve a successful outcome for the project. Table 9 lists these challenges and a description of each. Among these are issues directly related to content extraction discussed earlier (Challenges #1-5) as well as extraction issues for text sets that present special challenges and require special extractors to be trained to handle them (Challenge #6). An example of the latter might be text that is completely capitalized. This section addresses other challenges related to the interface needed for correcting the extracted information as well as database population.

Our system is designed around the extraction of assertions (relations/events) about people from unstructured text (e.g., newswire documents). These assertions can come from a batch of documents or a single document. In the case of a batch, there may be redundant repetitive data. For example, in a batch of documents about UK politics, many may state that Gordon Brown is a member of the Labour Party in Britain. Users may (1) want to be able to enter this information (with source) only once to allow them to concentrate on the entry of novel information or (2) enter this data many times for use as supporting information. If the user wants to enter the data only once, the system tries to first present the extracted candidates mostly likely to be accurate. This eliminates text snippets that have a pronominal reference since coreference chains can be determined with an F1 score of approximately 80% [13]. For example, “he is a member of the Labour Party” would be automatically be assigned a lower confidence score than “Gordon Brown is a member of the Labour Party.”

Another issue is that all events and relations are treated as facts by ACE. We prefer to label them assertions and to assign a source to the assertion, if provided. In a case like “Michael said” or “BBC reported,” Michael or BBC would be tagged as the source. When presenting content to a user in the interface, it is important for them to know the trustworthiness of a relation or event, and thus it is important in populating the knowledge base. So in this interface and database definition, a field for source attribution to an entity is available for all events and relations. Since no software existed for source attribution, code was written to provide this functionality.

**Table 9.** List of challenges

Challenge	Description
1. Accuracy of content extraction	The accuracy of extractors unaided is prohibitively low on key assertions.
2. Precision/recall trade-off	High precision means less User correction, but more manual entry. High recall means the reverse.
3. Current extractors lack appropriate inventory	47 data elements map to entities, relations, and events, but 20 elements are missing from or lack suitable ACE version.
4. User definitions don't match ACE	Definitions differ between terms in ACE and how Users define them.
5. Temporal anchoring and normalization	Extracting time tags and resolving them with source date. Some cases are complex or highly ambiguous.
6. Data specific challenges	Extracting from data sources that present special challenges when compared to the majority of texts.
7. Primary entity name selection	Detecting the most appropriate primary entity name, and allowing Users to choose primary entity names and effectively use them.
8. Disambiguating entities with KB	Assisting User in merging entity with existing record or creating a new one.
9. Accuracy of coreference resolution	Detecting if two phrases refer to the same entity, including a related issue of coreferencing “sets” of people.
10. Mapping from one schema to another	Converting User specification to extractor output and this output to the knowledge base data model.
11. Building successful interface & prototype	Developing an interface that allows easy entry, efficient correction and data verification.
12. Measurement of success	Measuring usability aspects of effectiveness, efficiency and User satisfaction.

The system is designed to allow users to set preferences to specify which types of data to extract, correct, and validate. For example, one user may be interested in social network information, such as family and friends. Another user might be interested in the travel of key sports figures.

Since extracted information is often incorrect, the system must also have a mode for correction and validation of information. This is done through drop down text menus in the hope that the user can quickly select a correct entity or relation. If the entity was missed by the extractor or misspelled due to author error, the user has the option to edit it. Since extractors frequently miss information [14], the system also requires a mode for entering information missed by the extractors.

The system must be able to select for any entity its primary entity name (PEN) from the list of names in the document (Challenge #7). As far as we are aware, there have been no evaluations or research on how easy this task is, but here we define the

PEN as the longest named mention of an entity in the document. For example, the PEN for David Beckham, the English footballer, would be “David Robert Joseph Beckham.” This name then has to be resolved with names currently existing in the knowledge base. The user has the option of creating a new entry for this name or adding the information to entities that already exist. Once this is determined, the system stores this name (and knowledge base id) in case it should arise in future documents. This software has been written but has not yet been evaluated. The user may want to change the display name to be different from the PEN, so the system has a method to allow for this and to link the PEN with the display name.

The user also has the option of stating a specific interest in information that appears about a particular person. Suppose a person is interested in Michael Jackson, the English footballer. Once Michael Jackson has a knowledge base identification number, the user can select him as an entity of interest (EOI). EOIs are separated in the user interface for the user. Documents contain many entities (typically about 50-100 per document) and if the user is only focusing on a few, having them as EOIs, makes them easy to find. Other entities that are assigned a knowledge base id, but not chosen as an EOI are referred to as resolved entities and are accessible from the interface as well. A resolved entity can be easily converted to an EOI, especially since it already has a knowledge base id.

The system stores a list of names that it has found in documents associated with a user’s EOIs. For Beckham it could have both “David Beckham” and “David Robert Joseph Beckham.” It compares these names with new documents to offer knowledge base ids for entities that have previously been seen by the system and assigned an id by the user.

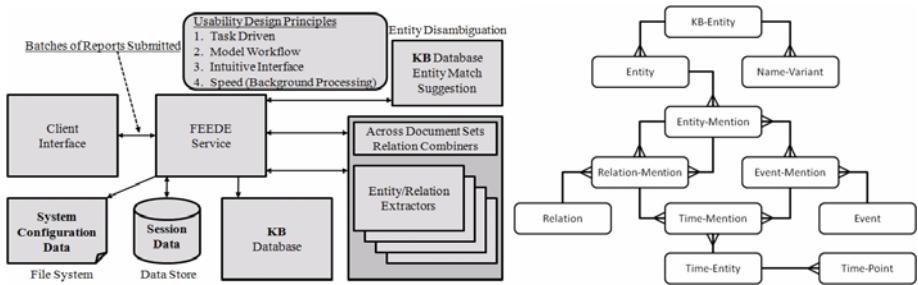
With a content related database, entity disambiguation is required when adding information from new documents (Challenge #8). If Michael Jackson, the English footballer, is an entity of interest, the user will have to determine that the information in the document being presented is his or her Michael Jackson. For example, “Michael Jackson” could be the singer, the American linebacker, the English footballer, the British television executive, or the former Deputy Secretary of the U.S. Department of Homeland Security.

The first time that the user has an entity Michael Jackson and goes to validate an assertion about Michael Jackson, the system returns the various named entities already in the knowledge base as well as any stored information that would assist the user to determine a match for this person. The user chooses whether to add the information to one of these existing entities or to create a new one. Selection or the creation of a knowledge base id is required for every entity in the assertion when the user chooses to validate an assertion as every entry in the knowledge base must have an identification number to determine its uniqueness.

Related to this disambiguation issue is entity coreferencing, an arena where extractors experience difficulty (good performance with names, moderate with pronouns, and poorest with nominals). There are also some inherent limitations with ACE in this regard, such as an inability to deal with sets of people (Challenge #9). For example, in “Ron met Joy after class, and they went to the store,” ACE cannot coreference “they” to “Ron” and “Joy.” This requires the development of software to extract possibilities other than those currently available.

All data is stored in an intermediary database before being uploaded to the main knowledge base. This allows the user to stop in a middle of a session and return before committing the data to the knowledge base (which is designed for manual entry). The architecture of our system that makes use of the FEEDE service is shown in Figure 1 (*left*). Additionally because the schemas used for ACE and the corporate knowledge base are different, they must be mapped to one another (Challenge #10).

Although extraction from unstructured text has low accuracy, extraction from the headers can be done with high accuracy. Extraction of items such as the document source, title, date and other key information save the user from manually entering this information. This data is extracted and presented to the user when uploading validated data. In particular, the document date is also required for temporal normalization, to resolve items such as “in February” if the year is not provided.

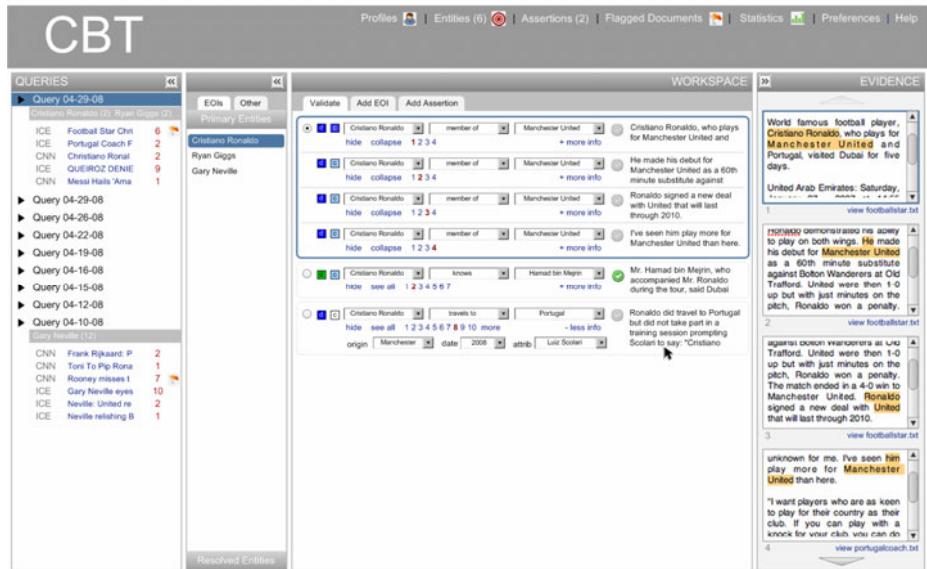


**Fig. 1.** System architecture (*left*) and overview of basic intermediate database structure (*right*)

The intermediary database is a relational database. When the documents are processed, the information extraction is stored in this database awaiting validation by the user. Thus it is populated with entities, relations, events, and time expressions, each with varying degrees of confidence. It is not intended for long-term but rather temporary storage of extracted information from documents so that a user can verify, correct, and validate content extraction results. Only the user validated or entered results are used to populate the main knowledge base after the user chooses to upload them to the knowledge base. A basic schema for the database is shown in Figure 1 (*right*). Links without prongs indicate “has one” while crow’s feet indicate “has many.” Entity-mentions have one entity, relation-mentions have two entity-mentions, and event-mentions and entity-mentions have a many-to-many relationship. Following this schema, each entity entry in the knowledge base (KB-Entity) can have any number of entities (and variants on its name) associated with it across document sets. Each of these entities in turn can have any number of mentions in a document. Each mention can be part of any number of relation-mentions or event-mentions, each of which refers to a single relation or event. Relation and event-mentions can also be associated with any number of time-mentions, each referring to a single time-entity, grounded to a timeline with defining time-points.

## 5 Interface Design

The correction/validation interface (a mock-up shown in Figure 2) that lies atop the intermediary database must allow for easy entry of missed information as well as efficient correction and verification of extracted data. The interface must make it as easy as possible for users to enter items not found by extractors into the database. This requires an examination of user workflow and the minimization of time required for the critical steps in extracting assertions (Challenge #11). Of paramount importance here is maximizing the efficiency, efficacy, and satisfaction of the users with the interface (Challenge #12), three properties of usability that should be independently considered [15].



**Fig. 2.** Example mock-up of the correction interface where a relation between two entities is selected and different mentions of this relation are presented to the user with corresponding textual evidence in the panels to the right. Note: The actual interface exists, and its appearance is very similar to this mock-up.

This interface has the following basic flow of content that is extracted from a corpus of documents. Because information presented to the user is “entity centric,” meaning that a user specifies entities of interest (EOIs) and the interface provides the relevant entities as well as related relations and events that involve those entities, the first part of this content flow from a set of documents is the list of entities (EOIs if the user has already specified these). Additionally, taken from a list of entities that appear as arguments to relations and events involving the EOIs, a list of secondary entities is also populated.

When a specific entity is selected in the workspace, the flow progresses to information about the entity from the document corpus. This information requires user validation. These are relations and events, with associated fields for arguments and other attributes (time, source-attribution, et al). Each of these pieces of evidence is also given a confidence (from high to low depending on textual and source factors) and an indication of whether the content is already validated and present in the KB (the two boxes to the left of relations in Figure 2). The shading of the boxes allows the user to quickly scan the data. Dark blue in the first box indicates that the information is not present in the main knowledge base and dark blue in the second box indicates that there is high extractor confidence in that information, suggesting that the user might want to examine that item first for validation. Tracking and displaying the presence or absence of the information in the knowledge base is important, as the users are often only interested in entering new information.

In the second piece of evidence, “he” is coreferenced with “Ronaldo.” Because the user is validating at an entity level, the PEN is the name present for each field that represents an entity, not the entity mention’s referent in the particular piece of text. The user then verifies relations and events by checking the textual evidence to the right of the relations, as well as the larger context for each on the far right if necessary. In the cases where the referent in the text evidence does little to clarify who the entity is (as with a pronoun), then other mentions of the entity can be indicated, as underlined in the 2<sup>nd</sup> evidence example in Figure 3 (*left*).

Because there are potentially multiple mentions of the same relation or event in the document corpus, the user can specify whether to see one or all of these at once. Each item the user desires to have entered in the main knowledge base must be checked and validated (the check circle on the right of the text). This information can be ignored if incorrect or corrected to form an accurate relation/event. Each of the arguments to the relation/event must be present in the knowledge base. The interface is structured so that arguments to relations and events can be modified via drop-down menus or typing in text fields. Some of these are accessed by clicking “more info.” When the relation or event has been corrected, if necessary, and is present in the material, then the user validates it for entry into the knowledge base. Otherwise there is no validation or a different relation or event can be validated if corrections significantly changed the relation or event.

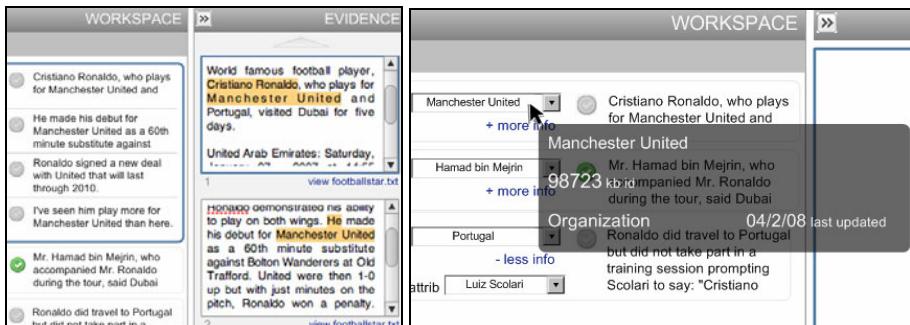


Fig. 3. Close-up examples of evidence (*left*) and the informative pop-up display (*right*)

Figure 2 also shows the implicit task flow as one looks from left (document sets and entity lists) to the right (extracted content to be corrected and textual evidence). The selected relation can be expanded (as in Figure 2 with the “member of” relation) to reveal the different instances of the relation in the text and the evidence that support it.

The interface also provides a convenient way to find out more information about entities, helping in disambiguating them. By simply leaving the mouse cursor over an entity in the workspace, the interface will generate a pop-up display about the entity, including information on its knowledge base identity, its type, and when it was last updated. This is depicted in Figure 3 (*right*).

As indicated, one can find in a rich knowledge base that there are many entities that have the same name and any entity that needs to be added must be linked to an existing entity or added as a new entity if not yet present in the knowledge base. In Figure 4, we present a close-up example of the disambiguation interface that reconciles an entity discovered in the text with other instances of the same name found in the database. Information about the entities is provided to help inform the user in determining which entity is the appropriate match. If none of these match, a new entity of the same name may be added.

	Hide	Hide	Hide	Hide	Hide
Alias	John Smith <input checked="" type="radio"/>	John Smith <input type="radio"/>			
Occupation	JS	Smitty	Rocket	Mr. Smith	
Place of Birth	Athlete	Pop Star	Athlete	Engineer	
Date of Birth	Tulsa, OK	Denver	London		Rome, Italy
Located at		May 30, 1980		April 16, 1975	
Member of					

**Fig. 4.** Close-up example of the entity disambiguation screen

## 6 Database Fields for Relations

Given the attributes present in ACE in addition to those we add to extend it, we can present a picture of what fields are necessary to store sufficient information in our database. As an example, the list of fields, along with definitions, for the relation-mention type follows. First we describe the fields necessary for the relations to interact with the knowledge base. Fields that must contain a value are marked as (R) for “Required.”

- account\_id (R) = Unique identification of the user looking at the relation, which is used to identify who validated and committed it.
- validated\_date = Time the relation was validated by the user.
- committed\_date = Time the relation was committed to the knowledge base by the user.

- modified\_date (R) = Last time the relation was modified by the user or an updated extraction.
- comments = Comments by the user on the information.
- user\_confidence = How certain the user is that this relation holds based on trustworthiness and ambiguity of the source text.

What follows next is a list of fields necessary for thorough relation-mention definitions in the database.

- doc\_id (R) = Unique identification of the document where relation was found.
- extractor\_info (R) = Features used to do the extraction, which determine the extractor used, the version, its parameter settings, etc.
- evidence (R) = Text of the document where the relation was found.
- evidence\_start (R) = Beginning index of the relation in the evidence.
- evidence\_end (R) = End index of the relation in the evidence.
- paragraph\_start = Beginning index of the paragraph snippet.
- paragraph\_end = End index of the paragraph snippet.
- arg1\_entity\_mention\_id (R) = Unique mention of the subject of the relation in the relevant document.
- arg2\_entity\_mention\_id (R) = Unique mention of the object of the relation in the relevant document.
- relation\_type\_id (R) = Uniquely determined by extracted type and subtype, this corresponds to a specific relation type that can be inserted into the knowledge base.
- tense = When the relation occurs with respect to the document. This can be past, present, future, or unspecified.
- attributed\_to = A reference to a person source or document source in the document for this relation mention.
- polarity = TRUE if assertion is so stated and FALSE when it does not hold (NOTE: this is not part of the ACE guidelines for relations). Most users prefer to enter positive assertions only, so the default value is TRUE.
- source\_confidence = Likelihood that this relation holds given conditional statements (e.g. may, believe, probably, etc.) and can be attributed to the writer of the document or some entity in the document or a combination of the two.
- extractor\_confidence = How certain the extractor is that this relation holds based on its own metrics.
- system\_confidence = A measure calculated by the system based on extractor confidence and its own internal knowledge.
- hidden = Set by the user, TRUE when the user wishes to hide the relation and FALSE otherwise.
- inKB = TRUE when the relation is in the knowledge base, FALSE otherwise.

With these fields, the relations are fully defined and prepared to be inserted into the final knowledge base.

## 7 Discussion

In this paper we have discussed a design for an interface to aid in the process of populating a knowledge base with corrected and validated knowledge originating from text sources that combines the benefits of both human users and automated extractors in order to make the process more efficient. We also examined the basic structure needed for the interface's underlying temporary database and the properties necessary for the information to possess in order to disambiguate and fully describe entities.

Also examined was the performance of two extractors, one rule-based and the other using statistical methods. The key point of this examination is to demonstrate that the extractors are very prone to errors and that in order to populate a knowledge base with what they produce a human being is still required to examine and validate text, even if facilitated by the interface and extractors' guidance. This interaction in the interface remains the most crucial element in ensuring that the process will be brisk, successful, and have minimal errors.

We expect that this system will save users significant time over manual entry of information into the knowledge base. Users have experimented with the prototype system and really like it. Their stated expectation is that this will save them lots of time. Our intention is to speed up the data entry process by 50%, which preliminary informal studies indicate is the case.

A GOMSL (Goals, Operators, Methods and Selection Rules Language) analysis was also performed [16] on the manual web client and our proposed system, which indicated that there should be a sufficient and significant advantage to using the new system.

Informal user observations were conducted on the first prototype of the FEEDE system with approximately 15 users. As a result, the following major changes were implemented to the system:

1. Waiting time minimization: the search of the knowledge base for alternative entities for disambiguation was time intensive and so parallel searches were implemented to decrease the waiting time for users.
2. Disambiguation screen redesign: the disambiguation interface screen had some modifications based on user feedback.
3. Potential recall correction: the extractor might have missed associating key items with an entity that users would want to enter into the knowledge base (e.g. date of birth). These unassociated entities are listed in the Entity section, which has been named Entity Explorer. The Entity Explorer has three main areas—unassociated, unresolved (no KB ID), and entity of interest (assigned a KB ID). Entities appearing in the unassociated category were correctly extracted, but could not be automatically associated with some relationship. A workflow was created to allow users to manually map the correct relationship in these “partial recall” situations.
4. Interface layout redesign: the evidence pane (rightmost pane of Figure 2) was redesigned. Horizontal scrolling problems due to source formatting caused inefficiencies for users trying to view the evidence, so it was redesigned to be a lower pane that filled the bottom of the screen.

5. Knowledge base resolver manager: this replaced the evidence panel on the right, allowing users to track items that required knowledge base identifiers.
6. Knowledge base verification: this functionality allows the user to view the items that they have uploaded to the knowledge base in order to check that everything was entered correctly.

A proposed change that has not yet been implemented is that of “Non-primary Relationship Assertion Highlighting”. User observations indicated that people would notice assertions about entities other than the one they were viewing and wonder if the content extractors were able to extract these items from the same document. The system was designed to group relationships in an entity-centric scheme, instead of document-centric, which could lead to questions about whether adjacent entity relationships were extracted or not, given the recall quality of the extractors. It was decided to highlight these items in grey, indicating to the user whether they would see this extraction for the other entity or whether they needed to manually enter it.

As further user studies and feedback are received, additional changes will be implemented to the interface and functionality. The ability to easily perform manual entry for items missed by the extractor is one such area yet to be addressed. Another proposed enhancement is to allow manual iterative refinement of the entity relationships in a manner that more closely matches the schema of the knowledge base, instead of the schema of the extraction system.

**Acknowledgements.** We would like to thank all of our team members and consultants for their advice and numerous contributions to this project.

## References

1. Grishman, R., Sundheim, B.: Message Understanding Conference – 6: A Brief History. In: Proc. 16th International Conference on Computational Linguistics (COLING), Ministry of Research, Denmark, Copenhagen, pp. 466–471 (1996)
2. ACE (Automatic Content Extraction) English Annotation Guidelines for Entities Version 5.6.1 (2005),  
[http://projects.ldc.upenn.edu/ace/docs/  
English-Entities-Guidelines\\_v5.6.1.pdf](http://projects.ldc.upenn.edu/ace/docs/English-Entities-Guidelines_v5.6.1.pdf)
3. Vilain, M., Su, J., Lubar, S.: Entity Extraction is a Boring Solved Problem—Or is it? In: HLT-NAACL – Short Papers, pp. 181–184. ACL, Rochester (2007)
4. Marsh, E., Perzanowski, D.: MUC-7 Evaluation of IE Technology: Overview of Results (1998),  
[http://www.itl.nist.gov/iaui/894.02/related\\_projects/muc/  
proceedings/muc\\_7\\_toc.html](http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/muc_7_toc.html)
5. ACE (Automatic Content Extraction) English Annotation Guidelines for Relations Version 5.8.3 (2005),  
[http://projects.ldc.upenn.edu/ace/docs/  
English-Relations-Guidelines\\_v5.8.3.pdf](http://projects.ldc.upenn.edu/ace/docs/English-Relations-Guidelines_v5.8.3.pdf)
6. ACE (Automatic Content Extraction) English Annotation Guidelines for Events Version 5.4.3 (2005),  
[http://projects.ldc.upenn.edu/ace/docs/  
English-Events-Guidelines\\_v5.4.3.pdf](http://projects.ldc.upenn.edu/ace/docs/English-Events-Guidelines_v5.4.3.pdf)

7. Working Guidelines ACE++ Events (2007) (unpublished Internal Report)
8. Automatic Content Extraction 2008 Evaluation Plan,  
[http://www.nist.gov/speech/tests/ace/2008/doc/  
ace08-evalplan.v1.2d.pdf](http://www.nist.gov/speech/tests/ace/2008/doc/ace08-evalplan.v1.2d.pdf)
9. Barclay, C., Boisen, S., Hyde, C., Weischedel, R.: The Hookah Information Extraction System. In: Proc. Workshop on TIPSTER II, pp. 79–82. ACL, Vienna (1996)
10. Donaldson, I., Martin, J., de Bruijn, B., Wolting, C., Lay, V., Tuekam, B., Zhang, S., Baskin, B., Bader, G., Michalickova, K., Pawson, T., Hogue, C.: PreBIND and Textomy—Mining the Biomedical Literature for Protein-Protein Interactions Using a Support Vector Machine. *BMC Bioinformatics* 4(11) (2003)
11. Ferro, L., Gerber, L., Mani, I., Sundheim, B., Wilson, G.: TIDES—2005 Standard for the Annotation of Temporal Expressions. Technical Report, MITRE (2005),  
[http://timex2.mitre.org/annotation\\_guidelines/  
2005\\_timex2\\_standard\\_v1.1.pdf](http://timex2.mitre.org/annotation_guidelines/2005_timex2_standard_v1.1.pdf)
12. Evaluation Scoring Script, v14a (2005),  
<ftp://jaguar.ncsl.nist.gov/ace/resources/ace05-eval-v14a.pl>
13. Harabagiu, S., Bunescu, R., Maiorano, S.: Text and Knowledge Mining for Coreference Resolution. In: Proc. 2nd Meeting of the North America Chapter of the Association for Computational Linguistics (NAACL 2001), pp. 55–62. ACL, Pittsburgh (2001)
14. NIST 2005 Automatic Content Extraction Evaluation Official Results (2006),  
[http://www.nist.gov/speech/tests/ace/2005/doc/  
ace05eval\\_official\\_results\\_20060110.html](http://www.nist.gov/speech/tests/ace/2005/doc/ace05eval_official_results_20060110.html)
15. Frokjaer, E., Hertzum, M., Hornbaek, K.: Measuring Usability: Are Effectiveness, Efficiency, and Satisfaction Really Correlated? In: Proc. ACM CHI 2000 Conference on Human Factors in Computing Systems, pp. 345–352. ACM Press, The Hague (2000)
16. Haimson, C., Grossman, J.: A GOMSL analysis of semi-automated data entry. In: Proc. ACM SIGCHI Symposium on Engineering Interactive Computing Systems, pp. 61–66. ACM, Pittsburgh (2009)

**PART III**

**Knowledge Management and**

**Information Sharing**

# Enterprise Wikis – Types of Use, Benefits and Obstacles: A Multiple-Case Study

Alexander Stocker and Klaus Tochtermann

Joanneum Research, Steyrergasse 17, A-8010 Graz, Austria  
Know-Center, Inffeldgasse 21a, A-8010 Graz, Austria  
`{astocker, ktochter}@know-center.at`

**Abstract.** In this paper we present the results of our explorative multiple-case study investigating enterprise wikis in three Austrian cases. Our contribution was highly motivated from the ongoing discussion on Enterprise 2.0 in science and practice, but the lack of well-grounded empirical research on how enterprise wikis are actually designed, implemented and more importantly utilized. We interviewed 7 corporate experts responsible for wiki operation and about 150 employees supposed to facilitate their daily business by using the wikis. The combination of qualitative data from the expert interviews and quantitative data from the user survey allows generating very interesting insights. Our cross-case analysis reveals commonalities and differences on usage motives, editing behaviour, individual and collective benefits, obstacles, and more importantly, derives a set of success factors guiding managers in future wiki projects.

**Keywords:** Enterprise 2.0, Web 2.0, Wiki, Knowledge Sharing, Case Study.

## 1 Introduction

Wikis, weblogs and social media platforms proved to be very successful on the Web. Including Wikipedia.com, Facebook.com, MySpace.com and many more they formed participative environments, allowing anybody to easily create, share and modify content with very limited technical expertise. Suchlike Web-2.0-platforms steadily lowered the barrier to share knowledge on the web and are nowadays rich sources for knowledge acquisition.

Motivated from their observations on knowledge sharing on the Web 2.0, enterprises have slowly begun to acknowledge the value of Web 2.0 principles and technologies. The corporate adoption of Web 2.0 was supposed to lead to manifold business advantages for various application domains. The ability of Web 2.0 applications, most notably wikis and weblogs, supporting both corporate knowledge workers and their practices, had been awarded with the term Enterprise 2.0 [15]. While weblogs may serve as a new media for corporate communication [13], wikis illustrate lightweight web based authoring tools supporting the collaborative creation of content in the enterprise [10]. Besides wikis and traditional weblogs two further types of Web 2.0 applications have emerged: social networking services [17] and microblogging services [1]. Both applications are also increasingly adopted in the Enterprise 2.0.

Cunningham defined a wiki as ‘a freely expandable collection of interlinked web pages, a hypertext system for storing and modifying information [and] a database, where each page is easily edited by any user’ [3]. The phenomenal growth of Wikipedia.com in both users and content inspired many organizations to experiment with own wiki-communities. Unfortunately, our literature review revealed that very few had been reported about the concrete use of wikis in the enterprise, yet: The International Symposium on Wikis (WikiSym) published just one paper on corporate wikis [14] in its five years history.

We reviewed the following contributions presenting empirical studies on enterprise wikis for our paper: Danis and Singer [4] conducted a longitudinal study of a wiki-based application deployed in a 900 member research organization. They found out that wiki-articles resulted in a greater transparency but as a technology the wiki not always provided fully appropriate affordances. Hasan and Pfaff [11] investigated a single case of wiki-rejection, thereby discussing many challenges and opportunities when adopting a wiki to manage corporate knowledge. Investigated management concerns dealt with flattening of organizational hierarchies and criticized the too innovative wiki approach towards knowledge acquisition. Explored social concerns dealt with the wiki-typical openness to vandalism, missing recognition for authorship and the poor quality assurance of wiki information. Surveying 168 corporate wiki users from different enterprises, Majchrzak, Wagner and Yates [14] revealed that enterprise wikis enhanced reputation, made work easier and helped the organization to improve its processes. Wikis particularly helped their organizations to improve workflows, increased collaboration efficiency and knowledge reuse and identified new business opportunities. Farell, Kellogg and Thomas [9] studied the use of wikis within IBM, requesting all IBM wiki owners to describe their benefits. They found out that wikis were primarily used as collaboration spaces for teams but also to support small ad-hoc groups as well as large communities and collectives. McAfee [15] investigated the use of wikis in the investment bank Dresdner Kleinwort Wasserstein, discussing the ability of wikis (as a portal) to replace email (as a channel) for certain issues, reducing information overload.

## 2 Research Design

### 2.1 A Multiple-Case Study Approach

The corporate adoption of wikis has rarely been analyzed in the academic literature. Benefits from intraorganizational wikis are – as well as obstacles – just starting to be explored. We still do not fully understand process, context and the specific phenomena to be observed when wikis are used in the enterprise [4]. This particular circumstance allows multiple-case study approaches to be very fruitful [8], [16] when aiming at the discovery of novel constructs to achieve theoretical advances.

We built upon a multiple-case study of three Austrian enterprises, adopting wikis to facilitate intra organizational knowledge transfer. Our three investigated enterprises operated in different environments, which may affect the conducted study in various ways and limits the comparison of the case-study results. The main goal of our paper

is to identify common patterns and differences across the cases. To understand the full context that is *how* and *why* benefits from the implemented wikis had been gained and *which*, our paper must provide sufficient information about the context, i.e. the starting point for the wiki, its implementation phase and the perceived value gain. We therefore intend to provide a detailed overview on our three cases. Table 1 summarizes their main characteristics of the three case companies – Alpha, Beta and Gamma. All three enterprises had completed the roll out of their wikis at least one and a half years before the start of our research.

**Table 1.** Key figures of investigated cases

	<b>Alpha</b>	<b>Beta</b>	<b>Gamma</b>
<b>Industry</b>	Micro-electronics	Engineering Services	IT-Services
<b>Number of employees</b>	2900	250	750
<b>Analyzed business unit</b>	Support Department	Whole Enterprise	Whole Enterprise
<b>Wiki users (potential)</b>	200	250	750
<b>Wiki users (estimated)</b>	70	180	100
<b>Years installed</b>	1,5	2	2
<b>Wiki purpose</b>	(Technical) Support	Technology, Workflows	Knowledge Base
<b>Wiki target group</b>	Support, R&D	All	All

In case *Alpha*, we explored the Austrian subsidiary of a large-scale multinational enterprise, developing highly innovative technical parts for automotive industry and industrial electronics. We probed an internal wiki-based solution implemented by the local support department. This solution was aimed to foster knowledge transfer within SD and beyond on the entire site, employing about 200 employees occupied with research and development.

In case *Beta*, we explored the Austrian subsidiary of a world-wide engineering group employing about 250 people delivering manifold engineering services. We probed an internal wiki conceptualized and implemented by a two person core-team responsible for knowledge management. The new solution was intended to support most notably technical project staff in knowledge documentation and learning within their periodic phases of low workload. Furthermore it should provide a central base for knowledge about processes relevant for the administrative staff.

In case *Gamma*, we explored a major Austrian IT service provider employing more than 750 people. We probed an internal wiki intended to serve as an electronic knowledge base in analogy to Wikipedia. The new solution was aimed to support everybody by providing stable, long-term knowledge, periodically required by all employees.

## 2.2 Data Collection and Analysis

Our multiple-case study uses quantitative and qualitative data to create a valid study, following the requirements of the respective scientific literature [8] enabling triangulation of evidence. We applied two data collection techniques:

Conducting structured interviews with 7 internal wiki experts in the first step, we asked them 40 questions about their perceived degree of organizational suffering requesting a new solution, their implementation strategy, and their perceived impact for individuals and organization, as differentiated in the (first) Delone and McLean model for information systems success [5]. All interviews lasted between two and three hours. We documented the qualitative empirical results in three case study reports sent to our interviewees to comment upon and ensure all details to be interpreted correctly, ensuring construct validity [18].

Responding to the request from academic knowledge management literature [12], we also emphasized on knowledge sharing from a non-executive employee's perspective. We therefore surveyed about 150 non-executive employees as knowledge workers [7], utilizing the deployed enterprise wikis in their daily business in a second step. The online questionnaire included 17 closed questions on reading and writing behavior, (knowledge) work practices, motivation for reading and editing articles, and perceived benefits and obstacles. In one case of very low wiki adoption, we requested additional information from non-wiki users. Analyzing the quantitative data collected through the survey, we compiled three 20-25 pages reports to guide executive employees in optimizing their wiki utilization.

Summarized, we wanted to find our, *how* and *why* enterprises wiki were used and with *what* results. We outlined the following guiding research questions for our study:

- How do enterprises use wikis to support employees in their daily business?
- Which motivation drives corporate knowledge workers to utilize wikis?
- What values are generated for individuals and the organization?
- Which success factors determine effective and efficient wiki-projects?

Each of these questions was analyzed highlighting the variety of answers across our three cases.

## 3 Multiple-Case-Study Results

### 3.1 Qualitative Results: Case Alpha

**Starting Point.** Because of the high degree of innovation of the conducted research, confidentiality was the utmost principle in the investigated enterprise. Therefore researchers operating in different project teams were separated from each other by entrance restrictions. The local support department, henceforth called SD, supported researchers and developers and provided guidance in all technical and methodical issues. Each department member was respectively responsible for one group of researchers. Because of the decentralized working environments, knowledge transfer within SD was not effective: Internal face-to-face meetings were very limited, yielding to heavy email-traffic and continuous reinventions of the wheel.

A wiki was considered to raise efficiency and effectiveness of SD's core responsibilities. The goal of this new solution was to facilitate knowledge transfer within SD and to raise the interconnectedness between department members. SD's manager expected the wiki as the most suitable platform for knowledge transfer, referring to the wiki-typical simplicity, its perceived acceptance as observed from Wikipedia, its special functionality, platform independence, and first and foremost the well-known wiki-principles, allowing every person to read and quickly edit articles at the same time. MediaWiki was favored as wiki software, because of its high degree of popularity and last but not least its proof of scalability.

**Wiki-introduction.** The Wiki was introduced top-down by SD's manager, who directly reported to the local site manager. By doing so, the wiki project received the important management commitment.

Respective MediaWiki-knowledge was available at the local site. No formal requirement engineering process was run through. First wiki properties and wiki structures had been eagerly discussed within internal group meetings, but no strict definitions arose. The creation of wiki articles should happen bottom-up, driven by the future users. A strong involvement of SD in content creation was supposed to lead to a lively wiki. Some relevant wiki content was also migrated from another repository, to assure immediate adoption. Although the wiki was based upon the requirements of SD, all employees at the local site were able to read and edit wiki content. Wiki users had to be logged in by providing their real names as anonymous editing was strictly forbidden, and only administrators were explicitly allowed to delete Wiki pages.

A series of actions had been taken to raise both awareness and acceptance of the wiki. The wiki was officially introduced within an SD jour-fixe. Furthermore, SD's manager personally demonstrated the usage of the wiki and its goals and forecasted benefits in other local departments. Relevant employees and opinion leaders were personally invited to actively participate and stimulate others to follow them.

The wiki allowed access to knowledge on tool-specific and methodical support for all in research and development. Researchers and developers should be able to focus their creative potential on the design of products. Applying wiki-knowledge, they could learn how to transform a quick idea into a commercial product.

Wiki-knowledge was organized by tasks and topics. Categories were used for meta-description and structuring of articles. When writing articles in the wiki, employees should avoid building too hierarchical structures. Such structures were supposed to unnecessarily increase complexity. An enterprise-wide roll out of the wiki as global support tool was cancelled, fearing the increase of complexity and information overload.

**Results after 1,5 years of Wiki Adoption.** Approximately 500 wiki-articles, periodically utilized by around 70 local employees, 15 of them highly involved in editing, had been created in one and a half years. Based upon a current server-log, the wiki had been accessed about 130.000 times since its roll-out and wiki articles had been edited about 10.000 times. These numbers signalize a very lively enterprise wiki.

The wiki was primarily intended to stimulate and foster knowledge transfer between SD members. However, it soon became apparent that even researchers themselves could benefit much from using the wiki. So far researchers were directly supported via face-to-face meetings, telephone-calls and emails by SD. Therefore researchers and

developers hesitated in active wiki-participation. From an individual perspective, it became more effective to directly request guidance from SD than to retrieve specific information from the wiki. While it was well known that researchers and developers always shared their knowledge on personal request, they lacked motivation to make their knowledge explicit in the wiki. Researchers even requested SD members to document ideas on behalf of them, stated doubts including “usage is very time-consuming”, “the wiki is too complicated”, “I am too lazy”, “I can directly ask SD”, or “I lack time”. The degree of raising the social or professional reputation by editing wiki-articles was perceived to be very low among researchers.

One important individual value gained from the wiki was the simple and easy to use full-text search, allowing quick guidance for emerging problems. Second, wiki articles incorporated formulations of both problems and their solutions on a very basic and therefore easy to understand level, which served the special needs of researchers very well. Another benefit dealt with the good level of transparency gained on support knowledge and respective knowledge holders.

The most important organizational value from the wiki was the rise of efficiency and effectiveness in SD’s core business, providing tool-specific and methodological support for researchers and developers. As a web-based solution the wiki ensured easy access without any special authorizations.

The following success-factors had been explicitly named by the corporate experts:

- A sufficient number of wiki-articles must exist right from start. Only then will employees perceive and accept the wiki as a useful knowledge base.
- The roll-out of the wiki must occur on a very broad user base and requires a handful convinced users stimulating others in personal face-to-face talks.
- The ‘built-in’ simplicity of the wiki-software is rather a minimum requirement than a success factor.

### **3.2 Qualitative Results: Case Beta**

**Starting Point.** As the enterprise was lacking an editorial intranet, documents and templates were mainly stored in complex hierarchical folders on file-system level or not accessible at all within a central database. These aspects limited the ability of employees to effectively document and share their technical knowledge.

In daily business, technical employees periodically returned to the headquarters from customer projects, using phases of low workload to prepare for upcoming projects. Prior to the wiki implementation, a lot of knowledge flew through the enterprise, because it was not successfully managed by any organizational or technical knowledge management tool. Another issue to cover dealt with managerial concerns: The management required a proper solution for documenting administrative processes within an electronic database to support the administrative staff.

A former senior manager of the enterprise was able to observe a successful wiki-implementation at a customer’s site, demonstrating documentation and sharing of technical knowledge in a very simple but effective way in analogy to Wikipedia. Reflecting on his own enterprise, he found a suchlike tool very advantageous for the project staff to explicate, codify and share their expertise. A wiki would enable technical project-staff to develop a knowledge base for all project relevant technical knowledge.

Based upon this initial situation, the main goal of the introduced wiki was to document all technical knowledge emerging from customer projects or elsewhere perceived to be useful for further projects in the future. Second, the wiki should be designed to document all process relevant knowledge to support the administrative staff.

**Wiki-introduction.** Perspective ([www.high-beyond.com](http://www.high-beyond.com)) was chosen as wiki-software: Simple WYSIWYG ('what you see is what you get') editing of pages, integrated file-system and document search, improved support of attachments, and active directory integration served as the main reasons for the selection.

While the implementation of the wiki had followed a top-down strategy driven by a department manager, the creation of articles was supposed to result bottom up. The wiki was divided into two sections: The first wiki section was dedicated to represent the knowledge of the technical staff – based on an enterprise-wide saying that 'all technical and organizational knowledge unable to be found via Google.com in less than two minutes' should be documented in the wiki. The second wiki section dealt with administrative issues and covered all various forms, templates and process descriptions.

The wiki had been implemented without external help by the two person wiki core-team, consisting of a technician and a sales representative. First wiki-structures and properties had been conceptualized in lively discussions with employees from various departments. While the core team was manually editing many wiki articles for administrative staff, only marginal content was collected to support technicians previously. All wiki-users were automatically logged in with their real names, not allowing any anonymous editing.

**Results after 2 years of Wiki Adoption.** From the perspective of the interviewed corporate experts, a wiki serves as an appropriate solution for knowledge transfer, documentation and sharing, if properly targeted. All 250 employees in the enterprise were able to both read and edit most of the wiki articles. Though, some sections, including administrative and project spaces had access restrictions due to confidential information stored therein.

About 180 employees utilized the wiki-knowledge provided in the technical section, consisting of about 500 wiki articles. However, only 15-20 employees coming from projects were able to use the wiki at the same time, i.e. document and share technical knowledge within the wiki, as access from customer sites was not supported. From studying wiki log-files the interviewed experts learned that on an average 15 wiki-articles were updated daily. Overall 20 technicians intensively created wiki-articles assuring a lively wiki with up-to-date technical knowledge.

The technical section had been strongly co-developed by the staff: In the beginning, some of them documented articles on a particular topic or technology having a private interest. But they soon realized the potential value of making their private knowledge professionally useable: As the wiki reflected all technical competencies of the enterprise, project managers were able to accurately acquire their project-staff based on the author-content relationship of wiki articles. It should also be noted that editorial efforts in the technical section were minimal, only dealing with the reassignment of articles to particular wiki-categories.

In contrast to that the administrative section was the problem child. Although intensive internal marketing activities had been conducted, the administrative staff hesitated to use the wiki and refused to update wiki articles. Most of the non-technical

articles had been created by a former wiki core-team member, who had left the enterprise. After his exit the up-to-datedness of wiki articles in the administrative section continuously declined, rendering most of them useless now.

Observing obstacles and barriers for wiki utilization, the core-team found out that technical staff was much more willing to ‘suffer’ from the additional work load triggered by the wiki. Non technical staff always complained about its lower comfort compared to their well-known office tools. Technical staff perceived a much higher value gain from using the wiki, most notably because of the faster and more structured access to project relevant technical knowledge. Articles within the technical section allowed not only access to textual content but also access to (software) tools located on file-system level. As an organizational benefit, the wiki simplified collaboration amongst (technical) employees. Technical staff successfully managed to use their idle capacities for knowledge sharing.

A huge obstacle accompanying the roll-out was the fact that employees only recognized the value of the wiki after having intensively used it. Unfortunately, communicating this very special aspect of social software to employees is extremely challenging. Any successful adoption of portals, like a wiki, must therefore be accompanied by a change in employee behavior. To achieve this change, a lot of management attention is required: Putting a ‘gentle pressure’ on employees might facilitate the emergence of effective wiki practices.

The following success-factors had been explicitly named by the corporate experts:

- Wikis require a dedicated core team in charge of all activities having reasonable time.
- Wikis require a corporate culture privileging openness and more importantly open communication.
- Management commitment and management attention are a must have: A company wide wiki may not be the initiative of a single person or department.
- Future wiki-users have to be integrated into conception and implementation right from the start.

### **3.3 Qualitative Results: Case Gamma**

**Starting Point.** Since the foundation of the company a plethora of internal databases partly containing redundant knowledge had emerged. Hence opinions were voiced demanding a more centralized environment. A 10 persons group responsible for knowledge management developed the idea to deploy a knowledge management tool building on the Web 2.0 principle user generated content. This group was very much attracted by the wiki philosophy, allowing everybody to contribute to a central database in a self organized way. They perceived Wikipedia.com as the archetype of their planned corporate wiki.

The aim of the introduced wiki was to develop a centralized electronic knowledge base involving all employees in content creation. The to develop company-wide encyclopedia was conceptualized to contain only a precisely defined set on topics and articles as well as the most prevalent abbreviations and short terms for products and services used in daily business. Such knowledge was not available in a centralized structure yet. Besides, the wiki should only preserve long-term knowledge to be accessed without any restrictions.

**Wiki-introduction.** The wiki had been introduced two years ago without external consultancy. However, some implementation support was provided by an affiliate company. JSP-wiki ([www.jspwiki.org](http://www.jspwiki.org)) was chosen as wiki software, as expert knowledge was available in the affiliate company. The wiki project team consisted of four selected members of the group responsible for knowledge management. The project team designed first wiki-structures and edited some wiki content. Intranet articles, flyers and news tickers were disseminated to facilitate the acceptance of the wiki. The wiki project was formally approved by the company management.

The wiki-group very strictly defined, which knowledge was allowed to be persevered in the wiki: basic information on customers, projects, technology, and expertise as well as information about the enterprise and the knowledge management group. The wiki contained glossaries, frequently used terms, project-names and explanations, descriptions of the departments, customer names and abbreviations. Meeting minutes, project relevant knowledge, knowledge related to interpersonal communication, news and specific reports were not intended to be part of the wiki as parallelisms of the wiki to the existing editorial intranet had to be avoided.

**Results after Two Years of Wiki Adoption.** Ten employees most notably managers as well as members of the knowledge management group take frequently use of the wiki. A second group, larger in number, perceives the wiki as a valuable tool but reflects that adopting such a tool affords a lot of voluntariness. Therefore, they rarely edit and only sporadically read wiki-articles. The largest group of employees does not use this wiki at all.

The project-staff responsible for the wiki introduction conceptualized the wiki as a fast-selling-item. But after two years of wiki adoption they learned that the majority of employees lack confidence in operating such a tool. Though the corporate culture was perceived to be very participative, employees sensed many obstacles to edit wiki content, most notably because of their lacking anonymity. Some employees had problems to understand the wiki-structure when trying to publish articles. However, surveying non-wiki users, we found that there are far more aspects slowing down the wiki success: Most of the wiki articles are merely relevant for the daily work assignments. Answering employees did not perceive an added value from the wiki. Furthermore, the aim of the wiki was perceived to be too broad and should be narrowed down.

Though wiki-users principally perceive a wiki as a valuable tool for their daily business, many of them hardly used it. They stumbled upon the challenging handling, most notably the uncomfortable wiki editor and the complicated wiki syntax. However, collecting and documenting information seemed to work acceptable from the perspective of the wiki group. But only few articles had been collaboratively edited, numerous wiki-revisions were only to be found on the portal pages. As an organizational benefit, the wiki increased the transparency on the organizational knowledge.

The following success-factors had been explicitly named by the corporate experts:

- Wiki-success requires the acquisition of first-movers stimulating others to participate.
- Wikis must be rolled out with sufficient articles motivating employees to participate.
- Though belonging to social software, wikis require very intensive internal marketing activities.
- Wiki users have to perceive the value of a wiki right from the start.

### 3.4 Quantitative Results

Surveying altogether 150 non-executive employees across all three cases, we were able to validate the results from the conducted expert-interviews. In this section, we present selected results on reading and writing behavior, type and frequency of wiki-contribution, sources of business-relevant information, motivation to read and edit articles, individual and collective benefits, and perceived obstacles of wiki adoption from a knowledge worker's perspective.

Knowledge about **reading and writing behavior** allows measuring the success of wiki implementations. Although the well known knowledge sharing dilemma [2] could be overcome on the Web, mainly due to the huge number of potential knowledge sharers, our study revealed that the situation in the enterprise is different: Reading behavior clearly differs across all three cases, but the relationship between reading and editing wiki-articles is similar: Only a very small fraction of employees counted for regular edits of wiki articles. Observed differences in wiki usage can be interpreted by referring on the different nature of our three cases. While *Alpha* and *Beta* demonstrated rather defined business-cases, stating wiki goal, context, target group and expected impact, *Gamma* remains more ambiguous as especially our survey of wiki deniers revealed.

The **lower editing behavior** in *Alpha* as compared to *Beta* can be explained by the precisely defined but smaller target group responsible for wiki articles in *Alpha*. The strength of *Beta* was the successful development of a lively enterprise wide wiki, as the high affinity of wiki users, most notably technicians, seemed to stimulate regular reading and editing practices.

Surveying on **type and frequency of wiki-contributions** we revealed that minor edits of existing articles and creation of new articles prevail. Correcting grammar and spelling, reverting articles using the revision history, restructuring articles and commenting articles were clearly outnumbered. Much of the special functionality of a wiki (e.g. revision history) remained unused across the three cases.

Surveying on enterprise-wide **sources of relevant information**, non-executive employees of *Alpha* and *Beta* clearer perceived the wiki counting to those. In *Gamma*, wiki-information seemed to bypass the demands of information seekers. Interestingly, employees of *Beta* seemed to prefer archives and portals including the Web, document-management and file-server towards channels, including telephone, email and face-to-face conversations. In *Alpha* and *Gamma* traditional media prevailed as source for business relevant information.

Finding business relevant information, facilitating one's individual work and observing what is happening within the enterprise accounted for the **main reasons to use** the wiki. To actively counteract email- and face-to-face-meeting overloads hardly stimulated wiki usage. However, such aspects were considered to come along with enterprise wikis in the literature [15]. Furthermore and contrary to the literature private issues still seemed to play a minor role in all three cases.

The **main motives** for non-executive employees **to actively participate** in article creation were the perceived value of their own wiki-contributions, the expectation of individual benefits from the wiki and the stimulation of colleagues to actively participate in content creation. As already known from the respective knowledge management

literature [6], reciprocity seemed to play a very crucial role along with wiki type knowledge sharing.

Surveyed on the **individual value gain** from wiki usage, non-executive employees in *Alpha* and *Beta* perceived the wiki had in some extent helped them to perform their business tasks quicker, facilitating their knowledge work. However, to a much lesser extent, they were able to raise their social and professional states. *Gamma*'s Non-executive employees seemed to be supported by the wiki to a lesser extent.

Surveyed on the **collective value gain** for team and/or organization, employees noticed an improvement of knowledge transfer and a boost in work performance in *Alpha* and *Beta*. In *Beta* the wiki also led to improved collaboration. However, the Wiki in *Gamma* seemed to generate only marginal advantages for the organization from a knowledge worker's perspective.

Surveyed on their **perceived obstacles** of wiki adoption employees identified aspects including few employees creating articles, few created articles, unequal write access, and time consuming editing and retrieval of knowledge. Interestingly, conflicts between wiki editors regarding the content of an article, and the transparency wikis entailed, were not considered to be major obstacles. Privacy issues seemed to play a minor role across all three cases.

## 4 Conclusions and Outlook

Investigating three different cases of enterprise wikis enabled us to gain many findings. Taking a closer look at the business perspective, our multiple-case approach revealed that enterprises still have a tough time when trying to map their business goals towards the goals of their wikis. Though enterprises understand the original benefits coming along with wikis as Web 2.0 type knowledge sharing tools, they still generate limited value for their employees. Case *Gamma* revealed best that just providing an enterprise wiki may not fulfill all the expectancies of the management. There is still a large gap between the knowledge management/sharing view and the business view. That gap must be overcome to fully exploit the potential of Enterprise 2.0. One commonality worth to mention is the fact that though all three wikis have been implemented top down the wiki article creation largely happened bottom-up.

We summarize that enterprise wikis have to solve a well specified knowledge problem which is crucial to the core business and relevant for the work practices of employees. Taking a non business perspective will limit enterprises to reason on a knowledge level as we found out when surveying corporate experts on wiki goals and benefits. Corporate experts mainly highlighted the soft benefits of wikis including generation of transparency on knowledge and deployment of a central and easily accessible knowledge base. From our studies we again learned that understanding the occurring business/knowledge problem and tackling it – in our three cases with a wiki – is the utmost principle in knowledge management. Our future work will aim to concretize differences between the business view and the knowledge view and suggest measures to overcome this gap. Table 2 summarizes the results of the multiple-case study.

**Table 2.** Multiple-case study results

	<b>Case Alpha</b>	<b>Case Beta</b>	<b>Case Gamma</b>
<b>Status quo</b>	Lacking knowledge transfer in support department	Lacking knowledge documentation and learning	Knowledge was not available in a central database
<b>Wiki goal</b>	Centralized and lively base for support knowledge	Easy documentation and sharing of technical and administrative knowledge	Centralized knowledge base
<b>Wiki Introduction</b>	Wiki for support staff, but also used by R&D ( <i>MediaWiki</i> )	Wiki for technical and administrative staff ( <i>Perspective</i> )	Wiki for everybody ( <i>JSP-Wiki</i> )
<b>Wiki Results</b>	Increased efficiency and effectiveness of R&D support Simpler search and retrieval of problem solutions	Facilitated technical knowledge sharing Better exploitation of phases of low workload	Improved collection and documentation of information
<b>Wiki Named Success Factors</b>	Provide sufficient wiki-articles right from start Roll-out on broad employee base Acquisition of convinced users motivating others	Dedicated and optimistic wiki team with reasonable time Corporate culture privileging open communication Management commitment and attention	Acquisition of first-movers motivating others Roll-out with sufficient wiki articles Performing intensive marketing activities

**Acknowledgements.** The Know-Center is funded within the Austrian COMET Program - Competence Centers for Excellent Technologies - under the auspices of the Austrian Federal Ministry of Transport, Innovation and Technology, the Austrian Federal Ministry of Economy, Family and Youth and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency FFG.

## References

1. Boehringer, M., Richter, A.: Adopting Social Software to the Intranet: A Case Study on Enterprise Microblogging. In: Proceedings of Mensch und Computer (2009)
2. Cabrera, A., Cabrera, E.: Knowledge Sharing Dilemmas. Organization Studies 23(5), 687–710 (2002)
3. Leuf, B., Cunningham, W.: The Wiki Way – Quick Collaboration on the Web. Addison-Wesley, New York (2001)
4. Danis, C., Singer, D.: A Wiki Instance in the Enterprise: Opportunities, Concerns and Reality. In: Proceedings of the ACM 2008 Conference on Computer Supported Cooperative Work, San Diego, USA (2008)
5. DeLone, W., McLean, E.: Information Systems Success: The Quest for the Dependent Variable. Information Systems Research 3(1), 60–95 (1992)
6. Davenport, T., Prusak, L.: Working Knowledge: How Organizations Manage What They Know. Harvard Business School Press, Cambridge (1998)

7. Drucker, P.: *Landmarks of Tomorrow: A report on the new 'post-modern' world*. Harper and Row, New York (1959)
8. Eisenhardt, K.: Building Theories from Case Study Research. *Academy of Management Review* 14(4), 532–550 (1989)
9. Farrell, R., Kellogg, W., Thoma, J.: The Participatory Web and the Socially Resilient Enterprise. In: *Proceedings of CSCW*, IBM T.J. Watson Research Center (2008)
10. Grace, T.P.L.: Wikis as a Knowledge Management Tool. *Journal of Knowledge Management* 13(4), 64–74 (2009)
11. Hasan, H., Pfaff, C.: The Wiki: an environment to revolutionise employees' interaction with corporate knowledge: OZCHI 2007 (2007)
12. Brent, H., Anantatmula, V.: Knowledge Management in IT Organizations From Employee's Perspective. In: *Proceedings of the 39<sup>th</sup> International Conference on System Sciences*, Hawaii (2006)
13. Kosonen, M., Henttonen, K., Ellonen, H.-K.: Weblogs and internal communication in a corporate environment: a case from the ICT industry. *International Journal of Knowledge and Learning* 3(4-5), 437–449 (2007)
14. Majchrzak, A., Wagner, C., Yates, D.: Corporate Wiki Users: Results of a Survey. In: *Proceedings of the 2006 International Symposium on Wikis* (2006)
15. McAfee, A.: Enterprise 2.0: The Dawn of Emergent Collaboration. In: *MIT Sloan Management Review* (2006)
16. Miles, M.B., Huberman, A.M.: Qualitative data analysis: A sourcebook of new methods. Sage Publications, California (1984)
17. Richter, A., Riemer, K.: Corporate Social Networking Sites – Modes of Use and Appropriation through Co-Evolution. In: *Proceedings of the 20th Australasian Conference on Information Systems* (2009)
18. Yin, R.: *Case study research: design and methods*. Sage Publications, Thousand Oaks (2003)

# A Knowledge Management System and Social Networking Service to Connect Communities of Practice

Élise Lavoué

Université Jean Moulin Lyon 3, IAE Lyon, Centre de Recherche Magellan  
Groupe SICOMOR, Lyon, France  
[Elise.Lavoue@univ-lyon3.fr](mailto:Elise.Lavoue@univ-lyon3.fr)

**Abstract.** Communities of practice (CoPs) emerge within companies by the way of informal discussions with practitioners who share ideas and help each other to solve problems. Each CoP develops its own practices, reinventing what is certainly being replicated somewhere else, in other companies. Our work aims at connecting CoPs centred on the same general activity and capitalising on all the produced knowledge. For that purpose, we propose a model of the interconnection of communities of practice (ICP), based on the concept of constellation of communities of practice (CCP) developed by Wenger. The model of ICP was implemented and has been used to develop the TE-Cap 2 platform. This platform relies on a specific knowledge management tool and a social networking service. We applied the model and platform to the case of university tutors. The TE-Cap 2 platform has been used in real conditions with tutors from different institutions and countries and we present the main results of this descriptive investigation.

**Keywords:** Community of Practice, Knowledge Indexation, Contextualised search, Social Networking, Web 2.0, Human–computer Interface, Online tutoring.

## 1 Introduction

Communities of practice (CoPs) emerge when practitioners connect to solve problems, share ideas, set standards, build tools and develop relationships with peers. These communities usually emerge within a company when people have informal discussions. Several communities interested in a same activity may exist but they can not know each other since they belong to different companies or are from different countries. They may develop similar practices without being necessarily aware of it. As a result, each CoP develops its own practices, reinventing what is certainly being replicated somewhere else.

Our work is illustrated throughout the article by the example of tutoring, which we define as the educational monitoring of learners during courses. Tutors usually belong to communities of practice within their institution. CoPs of tutors from different educational institutions prepare their own pedagogical contents for their students, and there is currently no possibility of reusing and sharing them. The result of this is that tutors lack help in their day-to-day practice, professional identity and practice sharing [1].

The problem which is challenging us is the creation of relation between CoPs of actors practicing a same activity so that they exchange their knowledge and produce more knowledge than separate communities. We aim at developing a Web platform to capitalise on all produced knowledge by contextualising it, so as to make it accessible and reusable by all members in their working contexts.

Our work is based on the concept of constellation of communities of practice (or CCP) developed by Wenger [2]. In this article, we first present the main characteristics of this concept, on which we base our research. We then situate our works by studying existing knowledge management systems and social networking services. In the third section, we propose a model of the interconnection of communities of Practice (ICP), as an extension of the concept of CCP. This model approaches the actors' activity according to the point of view of interconnected practices and considers CoPs' members to act as the nodes between CoPs to support knowledge dissemination. In the fourth section, we present the implementation of the model of ICP by the development of the TE-Cap 2 platform, meant for CoPs of educational tutors from different institutions, countries and disciplines who would tutoring. We finally validate our works by presenting the main results of a descriptive investigation.

## 2 Constellation of Communities of Practice

Explaining that some organisations are too wide to be considered as CoPs, Wenger sets out his vision of these organisations as constellation of communities of practice (or CCP) [2].

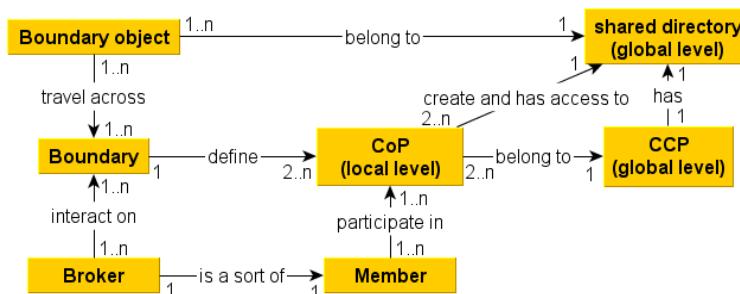
Communities of practice gather people together in an informal way [3] because of the fact that they have common practices, interests and purposes (i.e. to share ideas and experiences, build common tools, and develop relations between peers) [2],[4]. Their members exchange information, help each other to develop their skills and expertise and solve problems in an innovative way [5], [6]. They develop a community identity around shared knowledge, common approaches and established practices and create a shared directory of common resources.

We identify three main aspects of the concept of constellation, on which we base our works so as to develop a platform to support several Communities of Practice (CoP), summarised by Fig. 1:

- To favour interactions among CoPs. Brown and Duguid [7] brought the notion of “communities-of-communities” to develop the innovation within organisations, considering that the productions of separate communities can be increased by exchanges among these communities. The concept of constellation of communities of practice [2] resumes this idea by directing it on practices. The advantage to define several communities around shared practices is to create more knowledge and to develop more interactions than in a global community [5]. An involvement of this vision is to think about interactions among practices, rather than to favour information flows.
- To consider the boundaries of CoPs as places of creation of knowledge. The relations between communities can be supported by boundary objects [8] and by brokering. Boundary objects are products of reification and they constitute the directory of resources shared by all the communities. Interactions between communities relate

to this knowledge. “Brokers” belong to multiple communities and have a role of knowledge import-export between these communities. According to Ziovas and Grigoriadou [9], the combination of brokering as a product of participation and the boundary objects as a product of reification is an effective way to create relations between CoPs. The meetings on the boundaries of CoPs arouse interactions between the members, what makes boundaries the places of creation of knowledge;

- To establish a balance in the duality local/global. A person belongs to and involves in one or several CoPs, each bound to its local practices. But the concept of constellation approaches the CoPs in a global point of view, as a set of practices negotiated with only one shared resources repository. Every member, as broker, operates the dissemination of knowledge from a level of practice to another one. That is why it is necessary to supply all CoPs with multiple means of communication between practices which feed the shared directory [2].



**Fig. 1.** Modelling of the concept of Constellation of Communities of Practice (CCP)

### 3 Knowledge Management Systems and Social Networking Services

In this section, we situate our works with regard to KM systems and social networking services so as to show that we cannot use existing complete solutions.

A KM system has to support the KM process following three stages [10]: capturing knowledge, sharing and transferring knowledge, generating new knowledge. The KM platform of a company is aimed at its organisational entities, what implies that:

- These systems are not designed to CoPs which do not correspond to traditional organisational entities;
- The proposed computer tools are the only means for the employees to communicate remotely; they thus have to use them if they want to exchange their practices;
- The employees meet during meetings within their organisational entities, so weave relations except the platform.
- The employees belong to organisational entities for which they already have a feeling of membership.

Since our works concern actors who do not necessary belong to the same institution or the same company, we cannot use an existing KM platform. The most important difficulty to overcome is to arouse interactions between persons except any frame imposed by an organisation. For that purpose, it is necessary to bring them to become aware that they have shared practices and to provide the available means to get in touch with people from different CoPs.

Some Web 2.0 applications as Facebook or MySpace are social networking services which “connect you with the people around you”. They are very good examples of services which aim at connecting people who have common interests. Some social networking services are for more professional vocation, such as LinkedIn and Viadeo. But these sites are used for socialisation and to meet people. A consequence is that the tools offered to classify and to search for knowledge are not adapted to CoPs. Indeed, they often rest on collective categorisation in the form of tag clouds [11] (folksonomies) or on full text search. But this system of ‘tagging’ lacks structuring [12]. Within the framework of a CoP, we consider it is necessary to bring a knowledge organization to help users to index and search for knowledge. Tags systems work well for communities of interest where the users want to navigate within the application without precise intention. But these systems are not really adapted to CoPs where the users search for resources bound to working experiences. Users must be able to find a testimony, a discussion, an ‘expert’ or other resources (document, Web link...) very quickly, so that they can use it in their practice.

To sum up, we can use neither complete KM solutions nor existing social networking services but we can use existing components. We adopt one of the Web 2.0 principles: “innovation in assembly” [11]. When there are a lot of basic components, it is possible to create value by assembling them in a new way. We chose to develop a platform partially composed of existing Web 2.0 tools [13], available as well for KM systems as for social networking services, to capitalise knowledge and get in touch with people. Other part of the system consists of a knowledge indexation and search tool specifically developed to answer specific needs of CoPs, based on the model on the interconnection of communities of practice depicted in next section.

## 4 Model of Interconnection of Communities of Practice

The concept of CCP is based on the assumption that considering a global community as a set of interconnected CoPs increase member participation and creation of knowledge. Furthermore, this vision of an organisation takes into account as much the local level of every CoP as the global level formed by all the CoPs. We adopt this approach to develop a model of Interconnection of CoPs (ICP) which proposes to approach a general activity according to multiple points of view depending on actors’ practices. The development of the Web platform Te-Cap 2, depicted in section 5, is based on this model.

### 4.1 General Model of ICP

In the case of informal professions, such as tutoring, it is difficult to define exactly the field of practice of the actors. Actors’ activities can be seen as a set of different

practices which are similar in some points. For example, tutors' roles can be different as their interventions could be punctual or long-lasting; the learning session could be computer mediated or not and the learners' activity could be individual or collective. But some roles are shared by some of these contexts. We propose that this group of actors should be seen not as an endogenous entity defined by a field of practice, but rather as a set of CoPs supported by a Web platform where individual members acting as nodes of interconnected practices are the connection points (see Fig. 2). We suggest developing this concept that we have named Interconnection of Communities of Practice (ICP). This model aims at making existing local CoPs of actors (e.g. within an educational institution), who are engaging in the same general activity (i.e. tutoring), to get connected. This model also proposes active support for the dissemination of knowledge from CoP to CoP.

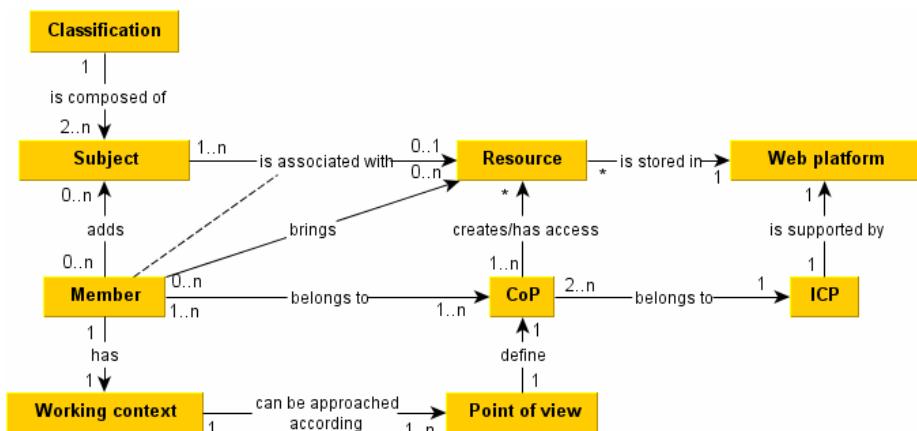


Fig. 2. General model of Interconnection of Communities of Practice

At an individual level, an actor's activity can be approached according to multiple points of view depending on the working context. In the ICP model, a CoP corresponds to the elementary level of actors' practice. The CoPs to which they belong are defined by their working context. At a general level, an ICP is composed of all the elementary CoPs defined by all the actors who participate in the Web platform. We could see it as a single community of actors practicing a same activity, brought together on the same platform; a group which can be approached from multiple points of view and accessed through multiple entry points.

For example (see Fig. 3), Tutor 1, working in the industrial engineering department of the University A in France who is monitoring a collective project about maintenance can belong to five different CoPs: tutors who monitor collective activities, tutors who are interested in maintenance, tutors who monitor educational projects, tutors of the industrial engineering department and tutors of the University A. Tutor 2 from another educational institution, for example University B in Canada, can belong to several CoPs, some of which Tutor 1 may also belong to.

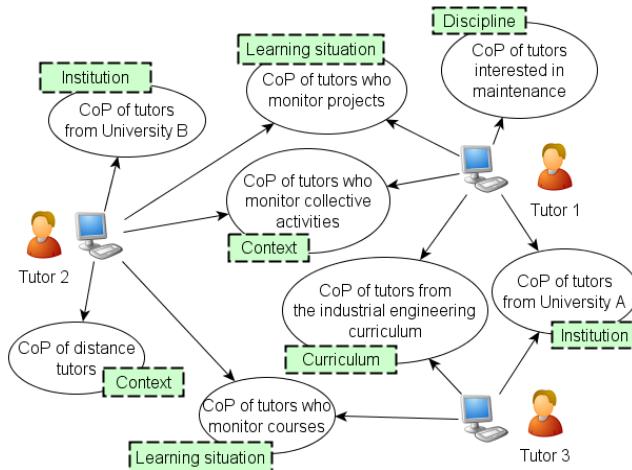


Fig. 3. Tutors as nodes of Interconnection of CoPs

These two tutors, from different countries, will be put in touch since their working context can be approached according to several similar points of view, which imply that they belong to same CoPs. Tutor 3 will be put in touch with both tutors because he belongs to the same educational institution and the same department as Tutor 1 and because he monitors the same type of activity as Tutor 2. So this example illustrates the fact that it is the tutors who are the nodes of Interconnection of CoPs. In this example, tutors' activity can be approached from several points of view: the context of the activity (collective, distance), the learning situation (project based learning, courses), the discipline (maintenance), the curriculum (industrial engineering) and the educational institution (universities). These points of view are categories of CoP and we propose in section 4.3 an approach to define a model of actors' practices, which implies determining all the categories of CoPs and which CoPs correspond to a given activity.

#### 4.2 The Reasons for Using ICP Instead of CCP

We based the model of ICP on the model of CCP since they suggest both considering wide organisations as a set of communities of practice which have common characteristics [2]:

- They share members: the ICP members belong to several CoPs, each corresponding to a point of view of their working context;
- They share artefacts: the ICP members participate on the same Web platform;
- They have access to the same resources: the ICP members have access to the shared directory of resources stored in the platform database.

However, an organisation defined as an Interconnection of CoPs (supported by a Web platform and composed of individual members who act as nodes of interconnected practices) does not form a Constellation of CoPs as defined by Wenger:

- Contrary to a CCP, the CoPs of an ICP do not share historic roots on which the mutual engagement of the members could base itself. The ICP members do not know apart the platform on which they join. This difference is fundamental because it raises the difficulty bringing persons who do not know each other to interact, what requires supporting a high level of sociability on the platform.
- In a CCP, the CoPs have interconnected projects which connect them whereas an ICP consist of actors practicing a same general activity who want to exchange on their practices with others, the community emerging by “propagation”. So that members are interested in the practices of the others, it is important to bring them to be aware that they have rather close practices which they can share.
- Contrary to a CCP, the ICP members do not belong necessarily to the same institution. Since we aim at supporting exchanges as well in members' local working context as at the general level of the activity, it is necessary that there are actors of various institutions.
- The CoPs of a CCP are in close proximity to each other, in particular geographically, whereas an ICP is constituted of persons who meet themselves on a Web platform and can thus be from countries of the whole world. This model does not thus include geographical proximity.

So, we propose a new model of ICP to represent a close but different type of organisation which could be seen as:

- An extension of the model of CCP in the sense that the conditions are less restricting. We showed that only three conditions on seven put by Wenger [2] are necessary to validate the existence of an ICP.
- A transposition of the model of CCP in the sense that it concerns persons gathered by a Web platform and not by a given institution or company.

### **4.3 Management and Dissemination of the ICP Knowledge**

The ICP resources are stored in a database according to a hierarchical classification composed of subjects based on a model of actors' practices. In the case of tutoring, resources correspond to explicit knowledge (documents and Web links) and tacit knowledge shared among members (e.g. exchanges of experience, stories, and discussions). We built a model of tutors' practices which defines at most four levels. The first level corresponds to the main factors which differentiate actors' practices (e.g. educational institution, curriculum, discipline, activity) and are the main categories of CoP. Each category is divided into subcategories and so on. The terminal nodes correspond to CoPs. This taxonomy of tutoring has been developed by an iterative process [14], based on interviews with six tutors (first development cycle) and on results of an experiment of a first prototype (second development cycle). The classification cannot be exhaustive because it is only a base which will evolve through modifications and additions made by the ICP members themselves.

When creating a resource (message, document, Web link), the author decide that it belongs to one or several CoPs by associating the name of the CoP (subject in the lowest level of the classification) with the resource. When they find a resource

(result of a search), members can also associate new subjects with this resource so as to spread it to new CoPs. They can either associate the name of a CoP to spread the resource to only a single CoP, or associate it to the name of a category of CoPs (subjects at higher levels in the classification) to spread the resource to all child CoPs. Indeed, Child CoPs (hierarchically lower level CoPs) inherit all the resources of a category of CoPs. So, ICP members' participation not only consists of creating new resources but also of creating links between these resources according to their relevance to the CoPs. This relevance is estimated by members themselves who consider a resource to be useful or interesting for a CoP. The supply of a resource to a CoP can lead to a debate on this resource and possibly to the creation of new resources for this CoP. Events reported in a precise context can lead to experience sharing (solutions, cases, scenarios), being used as a base to generate rules or recommendations which become global knowledge within the ICP.

## 5 The TE-Cap 2 Platform

We have developed the TE-Cap 2 (Tutoring Experience Capitalisation) platform according to a co-adaptive approach based on an iterative process including three development cycles. Each cycle rests on the development of a prototype, on its evaluation by the users by means of interviews or experiments and on the analysis users' activity [14]. This approach aimed at making users' needs emerge, at leading users to explicit these needs. The platform specifications evolved according to these emerging needs. We were particularly interested in developing a knowledge indexation and search tool for an ICP. We describe this tool in the following section.

### 5.1 User Profile Management

The knowledge indexation and search tool is based on the user profiles used to personalise subjects proposed to them. Users define their profile by filling several fields corresponding to categories of CoPs of the hierarchical classification. Values given to fields define CoPs and imply tutors' membership of these CoPs. The profile is composed of three main characteristics: identity profile, working context and secondary interests. The working context is about all the CoPs directly bound to actors' working context. The secondary interests are about all the CoPs which are not directly bound to their working context but which could interest them (give access to other resources able to interest them and to profiles of other people who share similar practices or experiences).

As a tool provided for the use of members of a CoP in their daily practice, this one offers them fast access to the relevant resources for them by two means (see Fig. 4):

- A link between the search interface and the profile allows users to only see the subjects from the classification which concern users and which interest them according to their profile. So users only have access to the resources of the CoPs to which they declare themselves to belong and can create resources only for these CoPs.

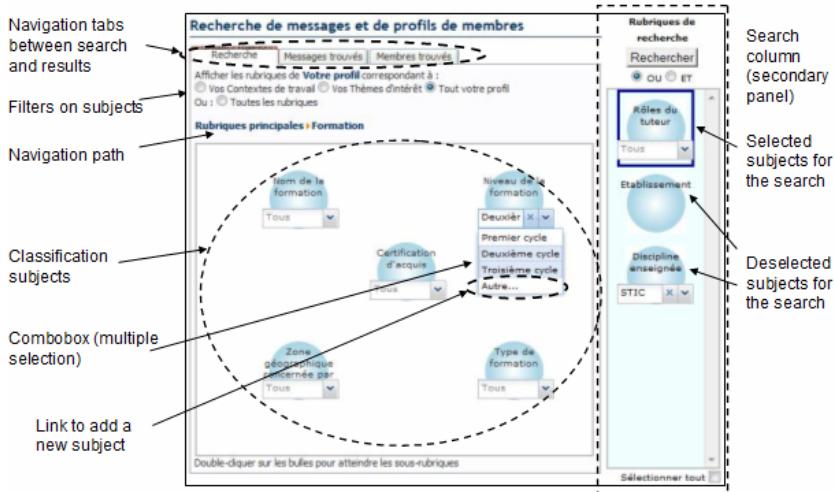
- Users have the possibility, according to their intention when connecting to the platform, to apply a filter to display on the classification interface only those subjects bound to their working context or to their secondary interests. In their daily practice, it is advisable to offer users at first only those subjects which concern their direct working context, this being the most efficient. If users do not find the information they look for in their direct working context, they must be able to extend the search to the other subjects of interest bound to their activity. In this manner they can find interesting ‘unexpected’ resources, which they can then bring into CoPs in which they have a central role.

## 5.2 Knowledge Indexation and Search Tool

The knowledge search and indexation tool, illustrated by Fig. 4, rests on the classification built for the ICP. The main panel (at the centre of the screenshot), composed of three tabs, allows easy and fast navigation between the results of the search and the classification. The tab ‘Search’ gives the possibility of navigating within the classification and of selecting search subjects. These subjects are represented in the form of bubbles, to bring conviviality and attractiveness to the interface. Users can navigate in the classification by a ‘double-click’ on a bubble which explodes it into more bubbles representing the sub-subjects. When reaching the last level (corresponding to the CoPs), subjects are represented in the form of a combo box allowing a multiple selection. Users can return to a superior level thanks to the navigation path. The platform proposes the same interface to search for posted messages and for member profiles, by separating them by the way of two tabs. In this way users can, at every search, consult the profiles of found members and ‘discover’ people who have similar practices or who offer expertise.

The secondary panel (on the right of the screenshot) gives the possibility of storing the subjects chosen for the search (by a drag and drop from the main panel). The subjects in this column are always visible when users navigate in the tabs of the main panel and from one request to another. Once in the “search column” users can deselect or select a subject (so as to refine or to widen the search), delete a subject by sliding the bubble outside the column and move bubbles inside the column to choose a preferred order. This principle of category selection can be compared to carts on commercial Web sites. This original human computer interaction has been chosen to promote navigation within the classification and to simplify the selection of items.

The indexing of an initiating message (starting a discussion) is made according to the following principle: users classify the message according to its context (bound subjects) at the same time as they write it. This principle aims at leading them to reflect upon the experience they relate. To facilitate this action, an interface in the form of tabs ensures an easy navigation, at any time, between the writing and the indexing of a message. The selected subjects in the classification column are then associated with the message, meaning that this resource belongs to the CoPs or categories of CoPs. Every user can associate the discussion with new subjects so as to spread the resource from one CoP to another one and from one level to another. Regulation is carried out by the author of the initiating message who has the right to remove the subjects which they do not consider relevant for the discussion.



**Fig. 4.** Knowledge search tool

### 5.3 Classification Evolution

Users can make the resource classification evolve through their participation on the platform, so as to lead to a classification using a vocabulary which gradually moves closer to the actors' practices. For that purpose, the interface gives at any time the possibility of adding a new subject to the classification, be it when filling in a profile, when classifying a resource, when searching a resource or when consulting a resource. The subjects used are recorded which allows for example the deletion of those considered useless. Unused subjects are later deleted, meaning that they were not adapted to the actors' field of practice or not located at the right level of the classification. This evolution of subjects is necessary so that the classification made a priori becomes closer to the reality of actors' practices and can follow the evolution of actors' uses and practices. It is also an important point for ensuring a coherence of all the CoPs forming the ICP and for offering a common identity to all the members.

## 6 A Descriptive Investigation

We conducted a descriptive investigation in real conditions, from 25 February 2008 to 5 July 2008. Our role consisted of encouraging registered tutors to participate by sending regular newsletters. The Web address of TE-Cap 2 was disseminated to several communities of tutors (ATIEF, t@d, PALETTE) and to virtual campus (VCiel, FORSE, E-Miage, Téluq, Master UTICEF, did@cTIC, FLE). We also sent an email to the users of the first prototype TE-Cap [14]. We wanted to develop the community around this existing core, hoping that they would encourage new users to participate. Discussion threads created during the first study were kept to be used as a base for new discussions. To help in the understanding of how the platform works,

we posted online demonstration videos: one general one and three specific ones (how to do a search, to write a message and to fill in the profile). This study aimed at testing the TE-Cap 2 platform as a support for the interconnection of CoPs of tutors. We defined indicators to measure sociability, levels of knowledge creation and sharing and utility of the platform. Results come from three types of data: use tracks (89 tables in the database with a total of 12732 recordings), thirteen answers to a questionnaire (thirty questions) and three usability tests.

Forty-two persons from nine francophone countries registered on TE-Cap 2. First of all, the answers to the questionnaire show that our aim to connect communities of practice of tutors from different institutions answers an existing need. Indeed, tutors look for information or practice sharing as much at the local level of their course (eight answers to the questionnaire) as at a more general level such as tutors' roles (twelve answers), technical and educational tools and resources (twelve answers), learners (ten answers) or learning scenarios (eight answers).

Although quite a few messages were written (fifteen) more users (twenty-seven) simply viewed discussions. This rather low activity can be explained by the fact that no tutor took on a leader role in the community life, inciting members to participate. According to questionnaires, people registered on TE-Cap 2 both to share experiences and practices and also to discover a new tool. This second reason implies a rather passive attitude and is certainly the cause of the lack of engagement in the community. Nevertheless, lurkers can also be considered as participants in a CoP platform. This group of people can become resource producers after a period of time. Also, the activity of reading is in itself an important part in a CoP development as well. As revealed by Chen [15], the mix of participation and non-participation shapes the identity of a community. Lurkers are often the majority in communities but they could be of great interest: 'heterogeneity in participation is to be expected, and it has its functions' [16].

A positive result is the rather large number of subjects added to the classification (forty-five), which implies a significant evolution in the classification and thus an appropriation by the users. The added values are coherent with the corresponding subject in the profile. But we observed no evolution (addition or deletion) of the subjects associated with a discussion thread. It is not a surprising result since the duration of the study was too short and the number of messages too low to observe the spread of a discussion from one CoP to another, or from one level to another.

Finally, usability tests carried out with three tutors according to a scenario, highlight the fact that the indexation and search interfaces of TE-Cap 2 are very easy to use and effective. But the use of these interfaces requires a learning stage, as is normal for an innovative interface which proposes new functionalities. Furthermore, users of the study did not see some innovative functionalities. One respondent's answer to the questionnaire confirms this point: 'According to your questions I perceive the potential of the platform'. Furthermore, twenty-three users did not fill in or did not use their profile which, we must assume, means they did not see the interest or did not take the time (it requires 5 to 10 minutes). The emphasised reason according to the questionnaire responses was that they did not understand the link between the profile and the proposed classification. It would be necessary to explain this link better so that they could see its relevance to their day to day practice (i.e. to filter subjects proposed for a search, according to their working context or interests). The help

brought by the videos was either not sufficient or not adapted (usability tests and use track analysis highlight the fact that when users connect to the platform, they do not watch the videos or just glance at them). An improvement could be the addition of a contextual help or a software companion.

## 7 Conclusions

In this paper, we defined a general model of the interconnection of communities of practice (ICP), based on the concept of constellations. This model aims at supporting knowledge sharing and dissemination for CoPs interested in a same general activity, in our case tutoring. We validated the implementation of this model by the development of the TE-Cap 2 platform. This platform was designed to connect several CoPs centred on same general activity and to manage their knowledge. The personalised interface offers users fast access to the relevant resources according to their working context. The knowledge indexation and search tool offers a structured and evolutionary method of knowledge classification. The dissemination of knowledge allows a learning and creation process of new resources for actors of different CoPs. The results of the descriptive investigation and usability tests tend to demonstrate the ease of use and the utility of the proposed tools and services, although not all the offered possibilities were taken up, as highlighted by use tracks. Further results will be obtained only by a use by a large number of persons and over a longer time period. It is only in these conditions that the platform and the proposed tools can be expected to reveal their potential.

The aim of this study was not to observe the emergence of an interconnection of communities of practice because it was unachievable in only four months. So as to observe such emergence, we plan to conduct another type of study, across a long-term period and with the addition of a software companion to facilitate the understanding of the innovative interface. It would also be interesting to address other communities than that of tutors or teachers who often tend towards rather individualistic professional behaviour and who are not always used to share.

## References

1. Garrot, E., George, S., Prévôt, P.: The development of TE-cap: An assistance environment for online tutors. In: Duval, E., Klammer, R., Wolpers, M. (eds.) EC-TEL 2007. LNCS, vol. 4753, pp. 481–486. Springer, Heidelberg (2007)
2. Wenger, E.: Communities of practice: Learning, meaning, and identity. Cambridge University Press, Cambridge (1998)
3. Lave, J., Wenger, E.: Situated Learning. In: Legitimate Peripheral Participation, Cambridge University Press, Cambridge (1991)
4. Koh, J., Kim, Y.: Knowledge sharing in virtual communities: an e-business perspective. *Expert Systems with Applications* 26(2), 155–166 (2004)
5. Pan, S., Leidner, D.: Bridging Communities of Practice with Information Technology in Pursuit of Global Knowledge Sharing. *Journal of Strategic Information Systems* 12, 71–88 (2003)

6. Snyder, W.M., Wenger, E., de Sousa, B.X.: Communities of Practice in Government: Leveraging Knowledge for Performance. *The Public Manager* 32(4), 17–21 (2004)
7. Brown, J.S., Duguid, P.: Organizational learning and communities of practice. *Organization Science* 2(1), 40–57 (1991)
8. Star, S.L., Griesemer, J.R.: Institutional Ecology, ‘Translations’ and Boundary Objects: Amateurs and Professionals in Berkeley’s Museum of Vertebrate Zoology, 1907–39. *Social Studies of Science* 19(3), 387–420 (1989)
9. Ziovas, S., Grigoriadou, M.: Boundary Crossing and Knowledge Sharing in a Web-Based Community. In: IADIS Web Based Communities Conference, Salamanca, Spain, pp. 248–256 (2007)
10. Von Krogh, G.: Developing a knowledge-based theory of the firm. University of St. Gallen, St. Gallen (1999)
11. O'Reilly, T.: What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software. O'Reilly Media, Sebastopol (2005),  
<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
12. Guy, M., Tonkin, E.: Folksonomies: Tidying up Tags? *D-Lib Magazine* 12(1) (2006),  
<http://www.dlib.org/dlib/january06/guy/01guy.html>
13. Wenger, E., White, N., Smith, J.D., Rowe, K.: Technology for Communities. Guide to the implementation and leadership of intentional communities of practice. CEFRIQ Book Chapter, pp. 71–94 (2005)
14. Garrot, E., George, S., Prévôt, P.: Supporting a Virtual Community of Tutors in Experience Capitalizing. *International Journal of Web Based Communities* 5(3), 407–427 (2009)
15. Chen, F.: Passive forum behaviors (lurking): A community perspective. In: 6th International Conference on Learning Sciences, Santa Monica, California, pp. 128–135 (2004)
16. Rafaeli, S., Ravid, G., Soroka, V.: De-lurking in virtual communities: A social communication network approach to measuring the effects of social and cultural capital. In: 37th Annual Hawaii International Conference on System Sciences (HICSS 2004), Big Island, Hawaii, pp. 70203 (2004)

# Design Issues for an Extensible CMS-Based Document Management System

João de Sousa Saraiva and Alberto Rodrigues da Silva

INESC-ID & SIQuant

Rua Alves Redol, nº9, 1000-029 Lisboa, Portugal

joao.saraiva@inesc-id.pt, alberto.silva@acm.org

**Abstract.** Content Management Systems (CMS) are usually considered as important software platforms for the creation and maintenance of organizational web sites and intranets. Nevertheless, a simple CMS alone typically does not provide enough support for an organization's more complex requirements, such as document management and storage. More specifically, such requirements usually present a set of design and implementation issues, which need to be addressed in an extensible manner if the system is to be maintained and evolved over time.

This paper presents the design issues that are subjacent to the architecture of WebC-Docs, a highly-customizable and extensible CMS-based web-application that provides document management functionality. Because of this degree of extensibility, the WebC-Docs toolkit can be configured and used in various kinds of scenarios.

**Keywords:** Document management system, Extensibility, Content management system, WebComfort.

## 1 Introduction

The worldwide expansion of the Internet in the last years has led to the appearance of many web-oriented CMS (Content Management Systems) [21,4] and ECM (Enterprise Content Management) [1,15,9] platforms with the objective of facilitating the management and publication of digital contents.

CMS systems can be used as support platforms for web-applications to be used in the dynamic management of web sites and their contents [3,17]. These systems typically present some aspects such as extensibility and modularity, independence between content and presentation, support for several types of contents, support for access management and user control, dynamic management of layout and visual appearance, or support for workflow definition and execution. On the other hand, ECM systems are typically regular web-applications that are oriented towards using Internet-based technologies and workflows to capture, manage, store, preserve, and deliver content and documents in the context of organizational processes [1]. Nevertheless, these two content-management areas are not disjoint [10]. In fact, it is not unusual to find a CMS system acting as a repository for an organization's documents and contents, albeit at a very "primitive" level, with problems such as: (1) no verification for duplicate information, (2) no grouping of documents according to a certain logical structure, and (3) no possibility of providing metadata for each document.

Into this context comes the WebC-Docs system. WebC-Docs is a document management toolkit for the WebComfort CMS platform [18], that provides a large set of configuration and extension points, which allows the system to be used in a wide variety of application scenarios.

In this paper we present the architecture of WebC-Docs and discuss its major technical contributions. This paper is structured in five main sections. Section 1 introduces the role of CMS and ECM systems as support platforms for web-applications, as well as the classical problem of document management in an organizational context. Section 2 presents the architectural aspects of the WebC-Docs system. Section 3 provides a brief discussion of this system. Section 4 presents related work that we consider relevant for this project. Finally, section 5 presents the conclusions for this project so far, as well as future work.

## 2 WebC-Docs

WebC-Docs [19] is a document management toolkit for the WebComfort CMS [17,18] with a component-based architecture, making it easily extensible, highly configurable, and adequate for several kinds of application scenarios.

This section presents some of the components and aspects of WebC-Docs' architecture, namely: (1) its key concepts; (2) its integration with the WebComfort CMS platform; (3) its Explorer-like web interface; and (4) some more technical features, such as: (i) the Document Versioning; (ii) the Dynamic Attributes; (iii) the Indexing and Searching; (iv) the Permissions mechanism; and (v) the facilities for using additional repositories to store and locate documents.

### 2.1 WebC-Docs' Key Concepts

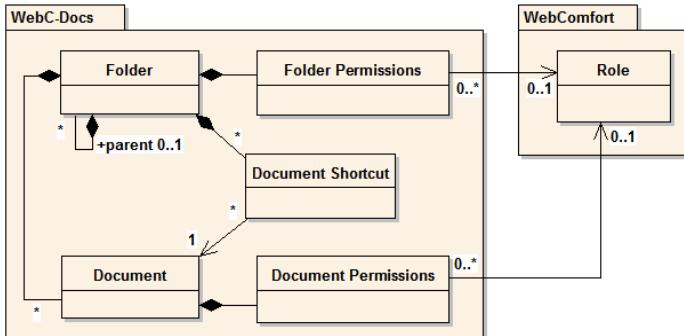
The WebC-Docs system consists primarily of the following concepts, illustrated in Figure 1: (1) Document; (2) Folder; (3) Document Shortcut; and (4) Document and Folder Permissions.

Document is the main entity; in a typical scenario, it represents a “real-world” document (the document’s digitized file may or may not be included in the Document; WebC-Docs allows the existence of Documents as simple “placeholders”, without the digital files of the real documents that they represent). On the other hand, a Folder consists of a Document container, but it can also contain Document Shortcuts, which are “pointers” to regular Documents. Thus, although a Document must be located in exactly one Folder, it is possible for the Document to be accessed from other Folders, by using Document Shortcuts.

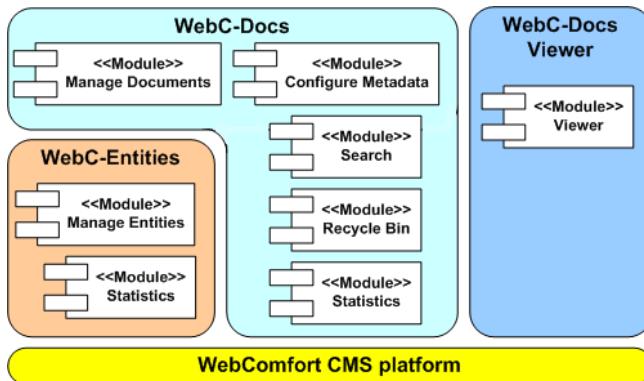
Finally, Document and Folder Permissions are associated with Documents and Folders, respectively, and specify what actions a WebComfort role can perform over them (this mechanism is explained further down this section).

### 2.2 Integration with the WebComfort CMS

One of the important innovations in WebC-Docs is its integration with a CMS platform, in this case WebComfort [17,18,20]. This integration consists mainly of the following



**Fig. 1.** The main concepts of the WebC-Docs system



**Fig. 2.** An overview of WebC-Docs' integration with WebComfort

points: (1) the user's main interaction points with WebC-Docs are implemented as WebComfort modules (e.g., Document Management, Statistics, Configuration) which take advantage of the facilities provided by WebComfort; (2) the Permissions mechanism (described below) uses WebComfort roles and users, instead of (re)defining those concepts; and (3) WebC-Docs is distributed as a WebComfort toolkit, and so it can be installed on any regular WebComfort installation in a simple and automatic manner. Figure 2 shows an overview of WebC-Docs' integration with WebComfort and other toolkits.

WebC-Docs provides a number of WebComfort modules that can be installed in WebComfort tabs (also called Dynamic Pages). These modules are the starting point for each of the system's use-cases. Subsequent steps – such as the visualization of the selected Document's Details – in those use-cases are typically handled by specific WebComfort Pages (not represented in Figure 2 for simplicity), by other WebC-Docs modules (when a Page would not make sense, such as deleting a Document and subsequently showing it in the Recycle Bin), or even by the module itself (when the use-case is very simple, such as moving a set of Documents to a different Folder).

Besides all this out-of-the-box functionality provided by the CMS, an added advantage of such an integration is that the system can be easily adapted to requirements



**Fig. 3.** The main WebC-Docs interface: folders and documents

such as those derived from an organization's structure or size (e.g., an organization's document management portal can consist of a select number of tabs – searching, management, configuration –, or it can provide a tab for each of its users, in which each tab contains the document management modules that are adequate for the user's responsibilities within the portal). For a description of WebComfort and its features, we recommend the reading of [17].

### 2.3 Explorer-Like Web Interface

One of WebC-Docs' main objectives is to be intuitive to the average user. To achieve this goal, we designed the main Document Management module to be similar to the typical file explorer interface (such as the one of Microsoft's Windows Explorer), as shown in the screenshot in Figure 3: the left side of the module displays the system's folder structure (always taking into consideration the current user's permissions), while the right side shows the documents contained within the selected folder.

The user can perform various actions over each folder and document (such as "Delete", "Move to", "Create shortcut to", and "Export to ZIP file"), which are displayed over the folder and document listings, respectively. Documents can also be downloaded immediately by clicking on the file type's icon, or the document's details page can be shown by clicking on the document's name.

WebC-Docs also provides a Recycle Bin (not to be confused with Microsoft Windows' own Recycle Bin), to which deleted documents will be moved. It will be up to a user with the "Documental Manager" role (which can be configured to be any one of WebComfort's roles) to regularly check the Recycle Bin's contents and purge it, or restore documents that have been deleted by mistake. The usage of this Recycle Bin can be disabled, although this is generally not recommended (because deleting a document would become an irreversible action).

### 2.4 Document Versioning

Document versioning is a fundamental aspect also addressed by WebC-Docs, making it possible to revert a Document back to the state in which it was at a certain point in time.

Each modification (e.g., a change to its description) does not really *alter* the document; instead, a new entry is created containing the new information, and the new file (if any) is stored in the Document's repository.

Reverting to a previous version will not erase all versions since then: a new Document version will be created, which will be identical to the version to which we are reverting. Thus, if a user later decides that the intended version V2 was one that was produced later than the version V1 (in which  $V2 = V1 + \text{a number of versions}$ ) to which we reverted, reverting to that version V2 is still possible.

Thus, a simple Document ID is typically not sufficient to be able to view and/or obtain a certain Document in WebC-Docs: a version number is also necessary. Nevertheless, to simplify the user's interaction with the system (as well as the referral to Documents via regular WWW links and bookmarks), if only a Document ID is specified, WebC-Docs automatically assumes that the user is referring to the latest version of the Document.

Document Versioning currently covers only Documents, their contents and their regular metadata (e.g., name, description, authors). Dynamic Attributes are not yet covered by this feature, but we will address this in the near future.

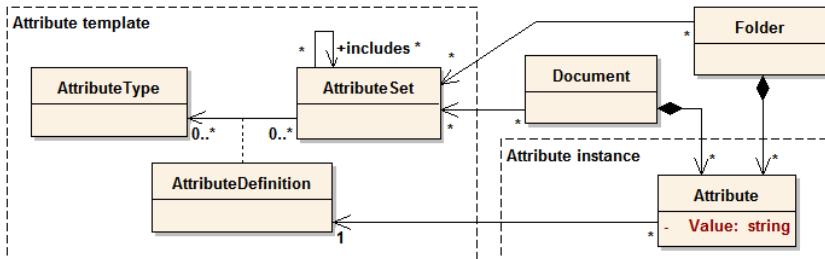
WebC-Docs offers Document Versioning as a configurable option, enabled by default. However, this option can be disabled if the Administrator so wishes (for reasons such as limited storage space on the document file repository, or no need to be able to revert documents to previous versions). Even if Versioning is disabled, the Document's history (dates of modifications, change-related comments) can still be viewed, as it is independent of Versioning.

## 2.5 Dynamic Attributes

One of the most powerful features of WebC-Docs is the possibility of specifying metadata at runtime, via the web-based interface, in a customizable manner that can be adjusted to the organization's document information requirements. This mechanism, which we designate as **Dynamic Attributes**, is based on the notions of Attribute Set, Attribute Definition, and Attribute Type, illustrated in Figure 4.

An Attribute Set consists only of a grouping of Attribute Types, and possibly even other Attribute Sets, allowing the specification of possible metadata using a tree-like structure. Attribute Type is responsible for defining the various kinds of attributes that can exist (e.g., integers, strings, dates, enumerations). This is done by pointing the attribute type to a specific class that must implement an interface with which WebC-Docs will communicate: this class will provide the various controls for viewing and editing attribute values. The associations between Attribute Sets and Attribute Types are called Attribute Definitions, and they are used to configure the Attribute Types in the context of the Attribute Set (e.g., name, default value, whether it is read-only).

Finally, this mechanism can be applied to Documents and Folders by using Attribute Sets and Attributes. A user can apply a Attribute Set to any Document or Folder, any number of times (e.g., to provide various author contacts for a certain Document). Attributes are simply used to store the values that are provided by the user.



**Fig. 4.** The main concepts of the “Dynamic Attributes” mechanism

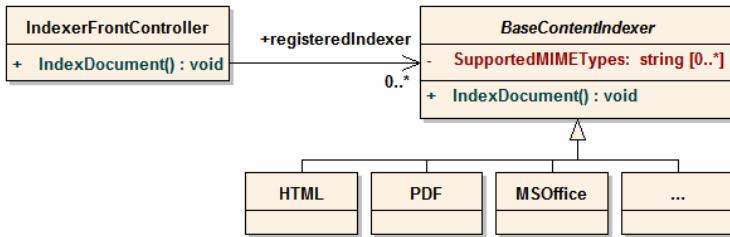
WebC-Docs currently provides out-of-the-box Dynamic Attributes for basic data-types (e.g., integers, strings) and enumerations (sets of strings to be used in selection lists). However, users with the appropriate permissions (e.g., Portal Administrator) can install new Dynamic Attributes by using the mechanism described above. An immediate advantage of this is that it enables the usage of any kind of attribute (e.g., a GIS-oriented Dynamic Attribute that displays a coordinate in a map, obtained via Google Maps).

## 2.6 Indexing and Searching

For a document management system to be of any practical use, it must allow its users to find documents given only some information about them (typically a string that is a part of the wanted document). WebC-Docs provides a WebComfort module with search functionality, using the Lucene.Net indexing engine [12], over all document information (i.e., regular document attributes, dynamic attributes, and document file contents). There are two types of search: (1) “regular search”, which operates like common search engines (if any part of a document matches the user’s search terms, the document is part of the search results); and (2) “advanced search”, in which the user can specify a variety of options – e.g., filter by dates, find all documents with a certain string in their file contents – to refine the search and reduce the number of false positives.

Figure 5 presents the main components of WebC-Docs’ indexing. Whenever a Document is created/altered, it is supplied as input to the `Indexer Front Controller`, which is responsible for quickly analyzing the Document (namely its MIME type and file extension) and forwarding it to the adequate `Base Content Indexer` objects (if any). Those objects will then parse the Document’s file contents and invoke WebC-Docs’ internal API (which, in turn, also provides a wrapper around the Lucene.Net functionality). Document metadata (regular document attributes and dynamic attributes) is always indexed, so the Document can still be found in the system, even if it has no file whatsoever.

The Searching mechanism itself is based on the “Pipes and Filters” design pattern [8]. Each segment/step of a search (filtering, searching) is performed by a single `Search Data Filter`; those search filters are combined into a “search pipeline”, at the end of which the search’s results are obtained. Although at first sight this approach may appear unnecessary (as the Regular Search actually uses only a single filter, and the Advanced Search uses two filters), the intent was to support new types of search in the



**Fig. 5.** Indexing can be performed over many file formats

future, likely with user-defined types of search (which, in turn, could be a composition of existing search filters) adjusted for the organization's information needs.

WebC-Docs also allows searching in the Document's own repository (if the repository allows it, of course), by means of the **Additional Repositories** mechanism, described further down this section.

Although WebC-Docs' indexing and searching functionality uses Lucene.Net, the user's search terms are not directly provided as input to Lucene.Net. Instead, WebC-Docs defines its own textual search language (similar to Google's own search engine), and uses a search term compiler built using CSTools [5]. This compiler is used to both validate the user search terms (e.g., too many parenthesis) and generate an AST (Abstract Syntax Tree) that will be provided as input to WebC-Docs' internal searching API (which will, in turn, transform that AST into a Lucene.Net query, and use that query to find documents in the various indexes that WebC-Docs uses). This intermediate search step provides two main advantages. First, it allows us to overcome some limitations of Lucene.Net, such as: (1) some efficiency issues when performing certain kinds of search (e.g., search without knowledge of the term's prefix characters), or (2) its inability to search for all documents *except* those that match a certain criteria (e.g., a search string like "NOT document" in WebC-Docs would return all Documents that do not contain the word "document" in them, but this search string would be considered invalid in Lucene.Net). Second, it provides additional pre-processing over the user search terms (e.g., to remove potentially dangerous words). Another possible advantage would be that, if we later decided to use a different indexing engine, the search language would remain the same.

## 2.7 Permissions

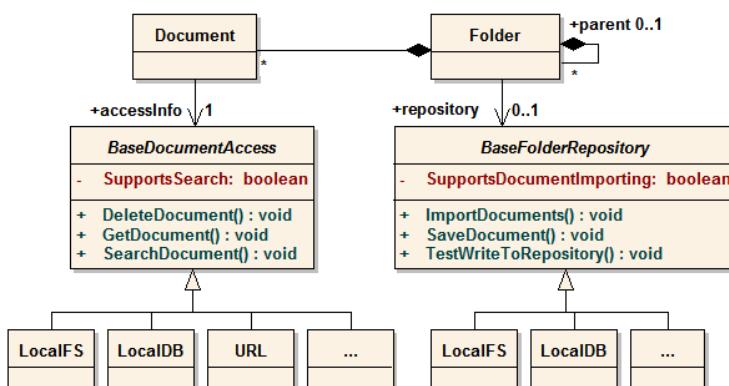
Permissions specification, perhaps the most important requirement in this kind of application, is addressed by WebC-Docs through the concepts of Document Permissions and Folder Permissions. This mechanism follows the typical ACL (Access Control List) philosophy, in which each of a user's operations (even viewing) performed over a document/folder are first checked for the user's permissions, to ensure that the user can perform that operation over the specified document/folder. To improve the system's usability, permissions are inherited (from parent folder to child folder/document) by default. For each of the roles defined in the WebComfort installation, a large variety of permission options can be configured (specifying permissions for anonymous users is also allowed, if the organization wishes to make some information publicly available).

The system can be configured to interpret permissions using either a **optimistic** or **strict** perspective. In the optimistic perspective, access is granted to a user over a certain folder/document if any of the user's roles has an explicit "allow" permission to it. On the other hand, the strict perspective follows a more traditional approach, by blocking access if *any* of the user's roles has permissions explicitly stating that access to the specified folder/document is blocked. This option allows the system to be easily adapted both to: (1) organizations that deal mainly with internal classified information (i.e., information that should only available to a few select users) and wish to enforce strict security validations regarding document access; and (2) organizations that intend to release most of their information to the public domain (and only some documents, such as works-in-progress, should not be publicly available).

Additionally, and considering that these permissions are likely to be accessed on a very frequent basis, WebC-Docs uses a "permissions proxy" mechanism. This proxy stores information about accessed permissions in the local server's memory, which accelerates future lookups of those permissions (e.g., for a user's access to a certain popular document). We say that this is a proxy (and not just a simple cache) because all permission-related operations (including modifications) go through it, removing the need to check for old or invalid entries.

## 2.8 Additional Repositories

Although the majority of WebC-Docs' usage scenarios will likely be based on the server's local file-system for document storage and a local database server for additional info (e.g., metadata), WebC-Docs can nevertheless use additional types of repositories for document storage and search. This can be useful for scenarios in which an external document repository already exists (e.g., a DSpace repository [6] in an academic context) and should be used by WebC-Docs. All of WebC-Docs' interactions with any repository are done through either Base Folder Repository or Base Document Access objects, shown in Figure 6.



**Fig. 6.** Support for multiple kinds of repositories

The main reason why repository access has been divided into those two classes, instead of using a single class containing repository access methods, is because all the information needed to access a document's file, or a folder's repository, is recorded both in the `Document` and `Folder` classes. This allows us to correctly handle cases in which a document, created in a folder  $F_1$  (that uses a repository  $R_1$ ), is later moved to another folder  $F_2$  (which uses a different repository,  $R_2$ ). This avoids moving files between repositories, which could itself present additional issues (e.g., the need to recover if a temporary repository failure is detected). This is also why search methods are found in the `Base Document Access`, instead of `Base Folder Repository`: to avoid contacting a repository regarding a search for documents that are not really there.

### 3 Discussion

Although we have presented some of WebC-Docs' important aspects, there are some key issues that should be mentioned. This section presents a brief discussion of such issues, as well as a few additional notes regarding the system.

A very important aspect in document management systems (in fact, in any kind of collaborative system) is *traceability*. WebC-Docs addresses this aspect in two complementary ways: (1) any action performed by a user (even an anonymous user) is recorded in WebComfort's own logging mechanism, along with all information that is relevant for that action, so that the CMS administrator can analyze that log; and (2) all document creation and modification operations are recorded in WebC-Docs' own historic records, which are not modifiable and can be viewed in the Document's details by anyone that has the required permissions.

WebC-Docs supports the specification of a Document's Authors and Editors. This is done by means of another WebComfort toolkit, WebC-Entities, that provides concepts such as `Entity`, `Person`, and `Group`. This allows WebC-Docs users to specify that "Person X is an author of this document", instead of just supplying a string of characters (which usually leads to name variations).

*Bootstrapping* is a typical problem in any system that involves setting permissions for resources; in WebC-Docs' case, no role has permissions to any Document or Folder, by default. The bootstrapping problem here is in ensuring that, although no particular role has access to a Document or Folder (by default), it is possible to configure Folders and Documents with non-default settings. This has been handled by making the CMS Administrator role (which *always* exists, and should be granted only to a few select users) be automatically granted access to any Document or Folder; thus, it is the CMS Administrator's responsibility to configure initial settings and permissions for WebC-Docs.

Regarding for the Dynamic Attributes mechanism, we consider the following notes relevant: (1) regarding the "Attribute template" section, it is not unlike the relations that the UML metamodel itself establishes between Class, Property, and Type [13]; and (2) regarding the "Attribute instance" section, it is not a case of linguistic instantiation, but rather ontological instantiation [2].

It is important to mention that WebC-Docs has already been validated in a number of case studies, namely: (1) the SIQuant [20] and WebComfort.Org [18] web-sites,

which were already using WebComfort, are now using the WebC-Docs toolkit to make available a variety of documents, such as white-papers and user manuals); (2) our own research group's document repository uses WebC-Docs to make our scientific production publicly available to the community [7]; and (3) WebC-Docs is currently being used by a Portuguese real-estate-related company, to organize and catalog their vast paper-based documentation archive.

Finally, it should be noted that, although WebC-Docs has not yet been integrated with other systems (e.g., DSpace [6]), it allows external sources to access search results by means of RSS [16]: a WebC-Docs search module can make its results available as a RSS feed, which can be consumed by an additional light-weight WebComfort toolkit called "WebC-Docs Viewer", or by any RSS feed reader (a regular reader such as those included with Microsoft Internet Explorer or Mozilla Firefox can view those feeds). Document details are made available in the feed by using the RSS extension mechanism, and the generated feeds are totally compliant to the RSS specification.

## 4 Related Work

Besides WebC-Docs, other document management systems already exist which handle some of the issues presented in this paper. In this section, we present some of which we consider most representative of this area, while making a comparison between those systems and WebC-Docs.

**OpenDocMan** [14] is a free document management system, distributed under the open-source GPL license, that is designed to comply with the ISO 17025 and OIE standard for document management [14]. Like WebC-Docs, it features a fully-web-based access mechanism (through a regular web browser), a fine-grained access control to files, and automated install and upgrades. OpenDocMan supports the concept of "transaction" because any document is in either the "checked-in" or "checked-out" states; WebC-Docs does not use this philosophy on purpose, because it would make it impossible for users to even *view* a document if it was checked out by a different user (e.g., because the user forgot to check the document back in). Also, like in WebC-Docs, OpenDocMan allows its administrator to add additional fields besides "document category" to further describe the document; however, those OpenDocMan fields are only strings, while WebC-Docs supports additional fields of any type (e.g., map coordinates) through its Dynamic Attributes mechanism.

**DSpace** [6] is an "open-source solution for accessing, managing, and preserving scholarly works" [6], designed to be as standards-compliant as possible. DSpace supports a tree-hierarchy, consisting of: (1) communities; (2) collections; (3) items; and (4) files. Although this hierarchy is fixed and cannot be altered, the user-interface can be adapted to "mask" some aspects of that hierarchy (e.g., to show a community as an aggregation of similar collections). However, this is a limitation that can make DSpace unsuitable for some particular (non-academic) cases – although it should be noted that DSpace's objective is to support the scholar community, and not a wider enterprise-like community.

It supports the addition of metadata fields to a resource-submission form, by adding a "name"—"type of HTML element to use for input" element to a textual file. Additionally, like WebC-Docs, its searching mechanism also takes advantage of the available

metadata (provided with each submitted resource) to provide more accurate search results. It should be noted that, unlike WebC-Docs, some kinds of changes to DSpace can only be done through the editing of textual files by an experienced administrator.

One of DSpace's primary goals is to handle the archive and preservation of documents. To this end, DSpace supports two different kinds of preservation: (1) *bit preservation*, which is like the typical file-storage mechanism (the document's bits are always the same); and (2) *functional preservation*, in which the document's contents are continuously adapted to the new formats, accompanying the passage of time and subsequent evolution of formats. Additionally, both contents and metadata can be exported at any time, using a XML-encoded file format. It also uses the Handle system [6] to provide unique identifiers for each managed resource, ensuring that document identifiers will be valid for a very long time.

**KnowledgeTree**'s document management system [11] features functionalities very similar to what can be found in current CMS systems (e.g., widgets); in fact, it could even be considered by some as a “document management system-oriented CMS”. However, unlike other web-based document management systems, KnowledgeTree's solution uses a rich-client (Windows-based) application combined with a web-application (installed on a web-server, and also providing a web-services communication layer), enabling a variety of useful operations (e.g., it allows drag-and-drop operations between local machines and remote document repositories). This rich-client perspective also allows integration with Microsoft Office applications (Outlook, Word, Excel, PowerPoint). On the server, it can index various file types, such as Microsoft Office, PDF, XML, HTML and text.

Its underlying concepts are very similar to WebC-Docs', namely the concepts of Folder and Document, which proves that the metaphor is adequate for this kind of system. It can have forums associated to documents, to enable discussions about certain documents, a feature that can easily be found in a CMS-based system (forums are a typical example of functionality offered by a CMS).

Its metadata-handling mechanism is also very powerful. It allows metadata of various types, which can be marked as “required”; nevertheless, WebC-Docs' Dynamic Attributes mechanism can also easily support these features. It also allows searching over metadata and over documents' contents (if they are indexed), like WebC-Docs. An interesting feature of this system is its concept of *document type*, which can be used to specify document categories, with associated metadata fields; while WebC-Docs does not support this kind of concept (because we believe that there can be many types of document, which are not mutually exclusive among themselves), it does support specifying the automatic application of an Attribute Set to new Documents, on a Folder basis. Additionally, it allows the usage of tags and tag clouds; WebC-Docs supports tags, but still only at a fairly basic level, because they can be easily replaced by Dynamic Attributes.

The system addresses document and metadata versioning, which allows a document to be reverted back to a previous point in time. As previously mentioned, WebC-Docs also supports document versioning, but only for regular metadata (e.g., authors, comments); Dynamic Attributes are not versioned yet.

Finally, we believe that one of WebC-Docs' greatest advantages over KnowledgeTree's system (or any “built-from-scratch” system) is its explicit usage of a CMS extensible platform (WebComfort): functionalities that are added to WebComfort can also easily be integrated with WebC-Docs (e.g., using forums, support for a Software-as-a-Service distribution mechanism), while specific systems must have such functionality created/adapted to their platform.

## 5 Conclusions and Future Work

The recent expansion of the Internet has originated many CMS and ECM systems that aim to facilitate the management and publication of digital contents. These platforms, which tend to be modular, extensible and versatile, can be used as support web applications for the dynamic management of web sites and respective contents. Nevertheless, most CMS platforms still do not offer functionality of a more complex nature (such as document management), while ECM platforms tend to address such complex functionality but without taking advantage of the possibilities that can be provided by a CMS platform.

In this paper we have presented WebC-Docs, a document management system for the WebComfort CMS platform. Besides typical file storage functionality (which can be found in many typical CMS installations), this system also provides features such as: (1) specifying customizable metadata for each document and/or folder; (2) the indexing and searching of documents using a powerful search engine, or even using additional search engines according to the folders where the search should occur; and (3) specifying fine-grained permissions for each document and folder, in a way inspired by traditional ACL mechanisms.

As for future work, we plan to introduce additional extensibility and configuration points to the system, in order to improve its adaptability to different organizational contexts. One of our priorities is adding an explicit configuration point to the system regarding the document indexers that are installed/used in the system; this will allow us to explicitly configure which indexers to use for each installation (e.g., if an organization wishes to index PDF documents using a particular PDF reading mechanism, then adding such a mechanism would be a matter of adding it through WebC-Docs' web-based interface).

Another aspect to further address is the integration with other document management systems, like DSpace. Although WebC-Docs supports the usage of additional data-stores, connectors have not yet been developed (only for local data-stores, such as the local file-system and a local Microsoft SQL Server database). Also, we intend to define a web-services interface, to allow other systems to interact with WebC-Docs without requiring its web-based interface.

Semantic relationships between documents is another issue that we intend to address. Currently, WebC-Docs does not allow “associating” documents with a certain relationship. Although this issue can be overcome by using folders and document shortcuts, it would be preferable to support user-defined document relationships (e.g., “document D1 is the result of the proposal in document D2”). This would also affect the search functionality, as document relationships could be analyzed in order to obtain potential search results.

Finally, an important topic that will be addressed in the very near future is the specification of document management workflows. Currently, WebC-Docs only allows the management of Documents and Folders, according to the current user's permissions. It would be desirable to specify the various steps of a workflow, in order to adapt to more complex application scenarios.

**Acknowledgements.** The authors would like to thank the members of EDM (Empresa de Desenvolvimento Mineiro, S.A.) for all their hard work, both in assisting with the testing of WebC-Docs in a real organizational environment, and in providing very helpful suggestions regarding the system.

## References

1. Association for information and image management (March 17, 2009),  
<http://www.aiim.org>
2. Atkinson, C., Kühne, T.: Model-Driven Development: A Metamodeling Foundation. IEEE Software 20(5), 36–41 (2009),  
<http://doi.ieeecomputersociety.org/10.1109/MS.2003.1231149>
3. Carmo, J.L.V.d.: Web Content Management Systems: Experiences and Evaluations with the WebComfort Framework. Master's thesis, Instituto Superior Técnico, Portugal
4. The CMS Matrix, <http://www.cmsmatrix.org> (retrieved December 9, 2009)
5. CSTools: Malcolm Crowe's Home Page (CSTools) (March 21, 2009),  
<http://cis.paisley.ac.uk/crow-ci0/>
6. DSpace: DSpace.org (March 22, 2009), <http://www.dspace.org>
7. GSI-Documents: INESC-ID, Information Systems Group (GSI) – Documents,  
<http://isg.inesc-id.pt/gsidocs> (retrieved March 21, 2009)
8. Hohpe, G., Woolf, B.: Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions. Addison-Wesley, Reading (2003)
9. Jenkins, T.: Enterprise Content Management Solutions: What You Need to Know. Open Text Corporation (April 2005)
10. Kampffmeyer, U.: ECM – Enterprise Content Management (March 17, 2009),  
[http://www.projectconsult.net/Files/ECM\\_WhitePaper\\_kff\\_2006.pdf](http://www.projectconsult.net/Files/ECM_WhitePaper_kff_2006.pdf)
11. KnowledgeTree: KnowledgeTree Document Management System (March 22, 2009),  
<http://www.knowledgetree.com>
12. LuceneNet: Lucene.Net (March 21, 2009),  
<http://incubator.apache.org/lucene.net/>
13. OMG: Object Management Group – Unified Modeling Language: Superstructure – Specification Version 2.0. (March 21, 2009),  
<http://www.omg.org/cgi-bin/apps/doc?formal/05-07-04.pdf>
14. OpenDocMan: OpenDocMan – Free Document Management Software DMS (March 22, 2009), <http://www.opendocman.com>
15. Rockley, A.: Managing Enterprise Content: A Unified Content Strategy (VOICES). New Riders Press, Indianapolis
16. RSS: Really Simple Syndication (RSS) 2.0 Specification,  
<http://blogs.law.harvard.edu/tech/rss> (retrieved March 22, 2009)
17. Saraiva, J.D.S., Silva, A.R.d.: The WebComfort Framework: An Extensible Platform for the Development of Web Applications. In: IEEE Computer Society (ed.) Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications (EUROMICRO 2008), pp. 19–26 (2008)

18. SIQuant: WebComfort.org,  
<http://www.webcomfort.org> (retrieved December 9, 2009)
19. SIQuant: WebComfort.org – WebC-Docs,  
<http://www.webcomfort.org/WebCDocs> (retrieved December 9, 2009)
20. SIQuant – Engenharia do Território e Sistemas de Informação,  
<http://www.siquant.pt> (retrieved December 9, 2009)
21. Suh, P., Addey, D., Thiemecke, D., Ellis, J.: Content Management Systems (Tools of the Trade). Glasshaus (October 2003)

# CrimeFighter: A Toolbox for Counterterrorism

Uffe Kock Wiil, Nasrullah Memon, and Jolanta Gniadek

Counterterrorism Research Lab, The Maersk Mc-Kinney Moller Institute  
University of Southern Denmark,  
Campusvej 55, 5230 Odense M, Denmark  
{ukwiil,memon}@mmtm.sdu.dk, jogni07@student.sdu.dk

**Abstract.** Knowledge about the structure and organization of terrorist networks is important for both terrorism investigation and the development of effective strategies to prevent terrorist attacks. However, except for network visualization, terrorist network analysis remains primarily a manual process. Existing tools do not provide advanced structural analysis techniques that allow for the extraction of network knowledge from terrorist information. This paper presents the latest work on the CrimeFighter toolbox for counterterrorism. The toolbox is designed based on past experiences working with investigative data mining, mathematical modeling, social network analysis, graph theory, link analysis, knowledge management, and hypertext. CrimeFighter consists of a knowledge base and a set of tools that each support different activities in criminal investigation work: data acquisition tools supporting web harvesting, knowledge structuring tools supporting information analysis, explorer tools for searching and exploring the knowledge base, algorithms for data mining, algorithms for visualization, algorithms for social network analysis, etc.

**Keywords:** Knowledge management processes, Tools and techniques, Counterterrorism domain, CrimeFighter toolbox.

## 1 Introduction

Knowledge about the structure and organization of terrorist networks is important for both terrorism investigation and the development of effective strategies to prevent terrorist attacks. However, except for network visualization, terrorist network analysis remains primarily a manual process. Existing tools do not provide advanced structural analysis techniques that allow for the extraction of network knowledge from terrorist information.

Theory from the knowledge management field plays an important role in dealing with terrorist information [1]. Knowledge management processes, tools, and techniques can help intelligence analysts in various ways when trying to make sense of the vast amount of data being collected. Several manual knowledge management processes can either be semi-automated or supported by software tools.

This paper presents the latest research on the CrimeFighter toolbox for counterterrorism. CrimeFighter provides advanced mathematical models and software tools to assist intelligence analysts in harvesting, filtering, storing, managing, analyzing, structuring, mining, interpreting, and visualizing terrorist information.

CrimeFighter is based on previous work from several research projects performed in the areas of knowledge management, hypertext, investigative data mining, social network analysis, graph theory, visualization, and mathematical methods in counterterrorism. Work on *iMiner* was targeted at constructing a framework for automated terrorist network analysis, visualization, and destabilization [2]. Work on ASAP (Advanced Support for Agile Planning) aimed at constructing a tool to assist software developers perform structural analysis of software planning data [3]. Finally, several projects have been performed to harvest terrorist information from the Web [4], [5], [6]. The important results from the above work are now being incorporated into the CrimeFighter toolbox.

The paper is organized as follows. Section 2 describes the knowledge management processes, tools, and techniques used by CrimeFighter to support the counterterrorism domain. Section 3 describes previous work relevant for CrimeFighter, while Section 4 outlines open issues. Section 5 presents ongoing and related work. Finally, Section 6 concludes the paper.

## 2 CrimeFighter Processes, Tools and Techniques

This section discusses how knowledge management processes, tools, and techniques play an important role for counterterrorism.

### 2.1 Processes

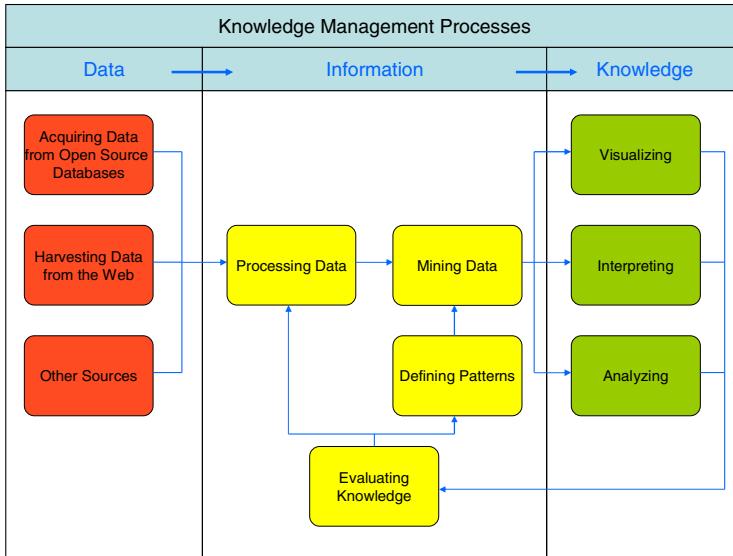
Several knowledge management processes are involved in the attempt to provide a toolbox that can support intelligence analysts in their work with terrorist information as shown in Figure 1.

Overall, the red processes involve acquiring data from various sources, the yellow processes involve processing data into relevant information, and the green processes involve further analysis and interpretation of the information into useful knowledge that the intelligence analysts can use to support their decision making.

**Data Acquisition.** Real intelligence data is hard to get due to its sensitive nature. In fact, very few researchers have been granted access to such data. Several options are available in the data acquisition processes:

- Data can be acquired from open source databases that contain authenticated information about terrorists and their activities. TrackingTheThreat.com is an example of a database that contains authenticated open source information about the Al Qaeda terrorist network. ([www.trackingthethreat.com](http://www.trackingthethreat.com)).
- Data can be harvested from the Web (including the dark Web – which is data not indexed by major search engines like Google, MSN, Yahoo, etc.). The Web contains many sources that potentially contain terrorist related information (i.e., regular Web pages, blogs, forums, search engines, RSS feeds, chat rooms, etc.).
- Data can be obtained from other sources such as databases maintained by intelligence agencies.

Our tools and techniques have so far been tested with open source data (the first two items above).



**Fig. 1.** Knowledge management processes for counterterrorism

**Information Processing.** The *Processing Data* step focuses on pre-processing of data. Data is cleaned from unnecessary elements and checked considering quality and completeness. The *Mining Data* step is concerned with processing of data using defined patterns (e.g., activities of people living or staying in the same city). Data mining algorithms are used in order to discover such hidden patterns and obtain relevant knowledge. The *Evaluating Knowledge* step is used to check whether the acquired knowledge is relevant. Errors are recognized and eliminated to improve the overall information processing. Possibly, new patterns are defined and old patterns are enhanced in the *Defining Patterns* step and the pre-processing of data in the *Processing Data* step is fine-tuned based on the feedback from the *Evaluating Knowledge* step.

**Knowledge Management.** The *Interpreting* knowledge step focuses on performing social network analysis in order to find new patterns and to gain deeper knowledge about the structure of terrorist networks. The *Analyzing* knowledge step focuses on supporting the work with emergent and evolving structure of terrorist networks to uncover new relationships between people, places, events, etc. The *Visualizing* knowledge step deals with the complex task of visualizing the structure of terrorist networks.

## 2.2 Tools

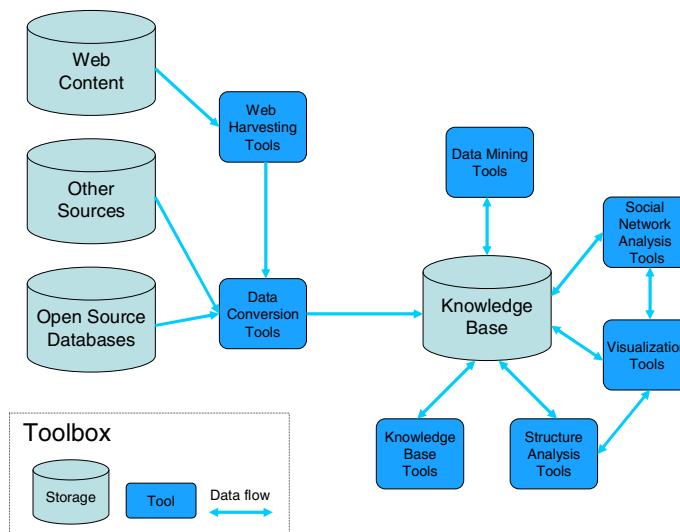
To support the knowledge management processes described in Section 2.1, CrimeFighter provides a number of tools. The toolbox philosophy is that the humans (intelligence analysts) are in charge of the knowledge management processes and the tools are there to assist the analysts. Thus, the purpose of the tools is to support as

many of the knowledge management processes as possible to assist the intelligence analysts in performing their work more efficiently. In this context, efficient means that the analysts arrive at better analysis results much faster.

In general, the tools fall into two overall categories:

- Semi-automatic tools that need to be configured by the intelligence analysts to perform the dedicated task. After configuration, the tool will automatically perform the dedicated task.
- Manual tools that support the intelligence analysts in performing specific tasks by providing dedicated features that enhance the work efficiency when performing manual intelligence analysis work.

The tools of the CrimeFighter toolbox are shown in Figure 2.



**Fig. 2.** Tools in the CrimeFighter toolbox

The heart of the toolkit is a knowledge base that contains information related to terrorism, which has been gathered and processed by dedicated tools. The content of the knowledge base is used by the various tools for further analysis and visualization.

The toolbox contains the following semi-automatic tools:

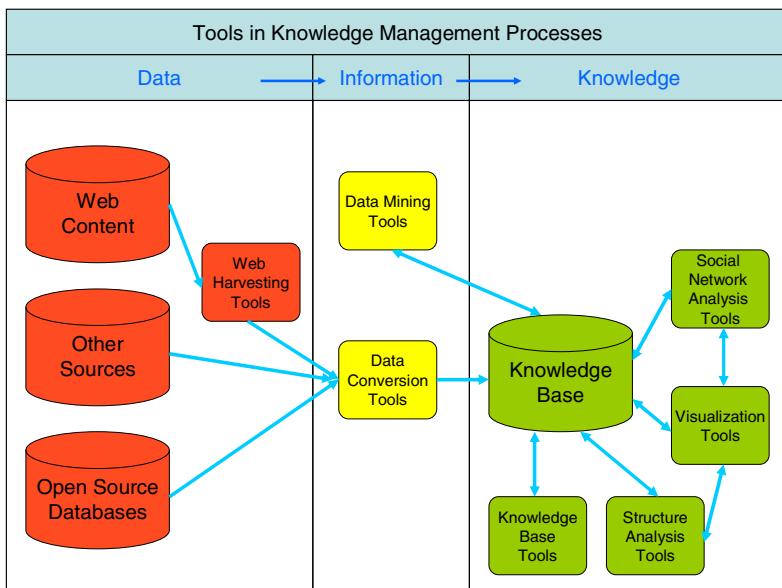
- Web harvesting tools make use of data acquisition agents (spiders) to harvest data from the Web. The spiders are controlled by the data conversion tools.
- Data conversion tools are responsible for both collecting (through spiders) and transforming data.
- Data mining tools provide selected data mining algorithms to discover new knowledge in data based on defined patterns.
- Social network analysis tools perform analysis to uncover new patterns and to gain deeper knowledge about the structure of terrorist networks.

- Visualization tools use graph layout algorithms to visualize discovered knowledge regarding terrorist networks. It can also be used as a graphics engine to support some of the tasks performed by the other tools in the toolbox.

The toolbox also contains the following manual tools:

- Knowledge base tools help maintain the knowledge base by allowing intelligence analysts to explore and revise the knowledge base content as well as to work with meta data.
- Structure analysis tools focuses on supporting the manual work with emergent and evolving structure of terrorist networks to uncover new relationships between people, places, events, etc.

Figure 3 shows how the different tools are related to the three overall knowledge management processes described in Section 2.1.



**Fig. 3.** Tools supporting the knowledge management processes

Some processes cannot be supported by tools and still have to be performed manually. The Evaluating knowledge step is an example of this. Intelligence analysts need to examine the quality of the knowledge and possibly alter the configuration of certain tools (i.e., data conversion, data mining, etc.) to obtain more relevant knowledge for their decision making.

### 2.3 Techniques

A number of advanced software techniques are used to develop the features of the tools (data mining, social network analysis, criminal geographic profiling, syndromic surveillance, hypertext, visualization, etc.). We will briefly describe these techniques to provide a better understanding of how they are deployed in our work.

**Data Mining** is a technique involving pattern-based queries, searches, or other analyses of one or more electronic databases, where a department or agency may conduct the queries, searches, or other analyses to discover or locate a predictive pattern or anomaly indicative of terrorist or criminal activity on the part of any individual or individuals [7]. Among the more prominent methods and tools used in data mining are [8]:

- Link analysis: looking for association and other forms of connecting among say, criminals or terrorists.
- Software agents: small, self-contained pieces of computer code that can monitor, retrieve, analyze, and act on information.
- Machine learning: algorithms that can extract profiles of criminals and graphical maps of crime.
- Neural network: special kind of computer programs that can predict the probability of crimes and terrorist attacks.

**Social Network Analysis.** The events of 9/11 instantly altered the perceptions of the words “terrorist” and “network” [9], and the United States and other countries rapidly started to gear up to fight a new kind of enemy. In conventional warfare, conducted in specific locations, it is important to understand the terrain in which the battles will be fought. In the war against terror, there is no specific location. As 9/11 showed only too well, the battleground can be anywhere. The terrorists’ power base is not geographic; rather, they operate in networks, with members distributed across the globe [10]. To fight such an enemy, we need to understand the new “terrain”: networks – how they are constructed and how they operate. Using techniques of graph theory and network analysis to analyze social networks, such as terrorist networks, a specialized sub-discipline known as social network analysis rapidly developed in the years leading up to 9/11 and has been a hotter topic since. The applicability of social network analysis to fight crime and terrorism had been known to specialists for many years, but it was only after 9/11 that the general public realized the critical importance of “connecting dots” in investigations and surveillance of terrorists [8].

**Criminal Geographic Profiling** is a technique originally designed to help police forces to prioritize large lists of suspects typically generated in cases involving serial crime [11], for instance, murder and rape [12]. The technique uses the location of related crime sites to make inferences about the most likely area in which the offender might live (or visit regularly), and has been extremely successful in this field [13], [14]. The need for such a technique arises because investigations of serial crimes frequently generate too many, rather than too few, suspects.

**Syndromic Surveillance** is an innovative electronic surveillance system (automated extraction and analysis of routinely collected data) which use data based on disease symptoms, rather than disease diagnosis [15]. It involves collecting and analyzing statistical data on health trends (such as symptoms reported by people seeking care in emergency rooms or other health care settings) or even sales of flu medicines. Because bioterrorism agents such as anthrax, plague, and smallpox initially present “flu-like” symptoms, a sudden increase of individuals with fever, headache, or muscle pain could be evidence of a bioterrorist attack [16]. By focusing on symptoms rather

than confirmed diagnoses, syndromic surveillance aims to detect bioterrorism events earlier than would be possible with traditional disease surveillance systems.

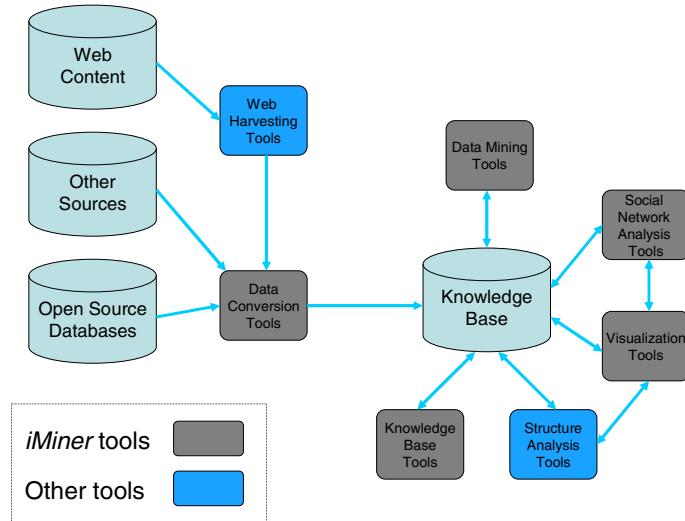
**Hypertext.** Organizing and making sense of information is an important task for intelligence analysts and has been the main focus of hypertext research from its very beginning. Hypertext systems aim at augmenting human intellect – that is “increasing the capability of a man to approach a complex problem situation, to gain comprehension to suit his particular needs, and to derive solutions to problems” [17]. The most widely used structure abstractions in hypertext are nodes and links. Nodes are informational units that can be connected through links. Users can traverse links and thereby navigate through a hypertext (graph). Nodes and links, however, have been criticized for a lack of support for emergent and evolving structures. Spatial hypertext was designed for and is well suited for dealing with emergent and evolving structures [18]. Thus, hypertext theory (in particular spatial hypertext theory) plays an important role for the structure analysis tools.

**Visualization.** Information synthesis and analysis can be facilitated by a visual interface designed to support analytical processing and reasoning. Such an interactive visualization approach is also known as visual analytics [19]. Visually analyzing social networks has been receiving growing attention and several visualization tools have been developed for this purpose. *Vizster* [20] provides an environment to explore and analyze online social network, supporting automatically identification and visualization of connections and community structures. *SocialAction* [21] allows users to explore different social network analysis measures to gain insights into the network properties, to filter nodes (representing entities), and to find outliers. Users can interactively aggregate nodes to reduce complexity, find cohesive subgroups, and focus on communities of interest. However, the measures used in these systems are topological-oriented. A framework for automatic network analysis and visualization was proposed in [22]. Their *CrimeNet Explorer* identifies relationships between persons based on frequency of co-occurrence in crime incident summaries. Hierarchy clustering algorithm is then applied to partition the network based on relational strength. A visual analytic system *Jigsaw* [23] represents documents and their entities visually in multiple views to illustrate connections between entities across the different documents. It takes an incremental approach to suggest relevant reports to examine next by inspecting the co-occurred entities.

### 3 Previous Work

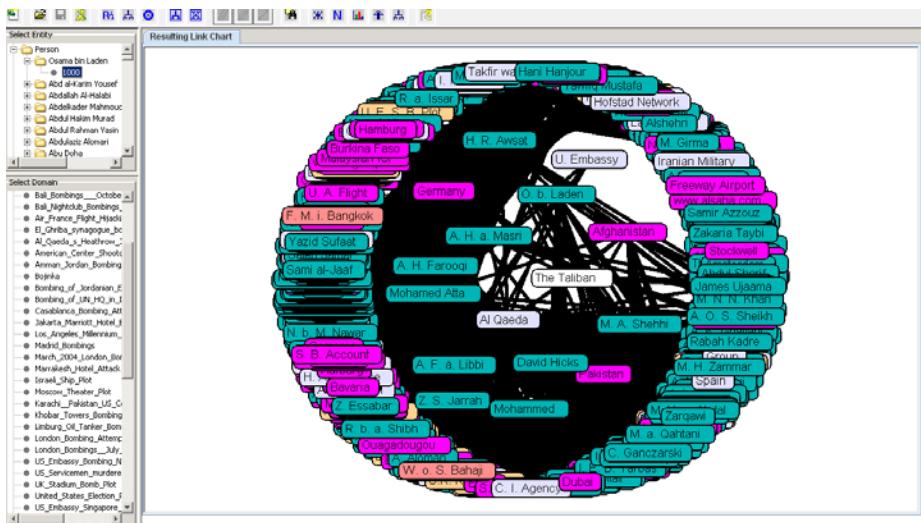
This section briefly describes our previous work relevant for the CrimeFighter toolbox. Additional detail can be found in the provided references. Currently, many of the identified knowledge management processes for counterterrorism are supported by our tools. Figure 4 shows our previous work.

The *iMiner* prototype includes tools for data conversion, data mining, social network analysis, visualization, and for the knowledge base. *iMiner* incorporates several advanced and novel models and techniques useful for counterterrorism like subgroup detection, network efficiency estimation, and destabilization strategies for terrorist networks including detection of hidden hierarchies [2].



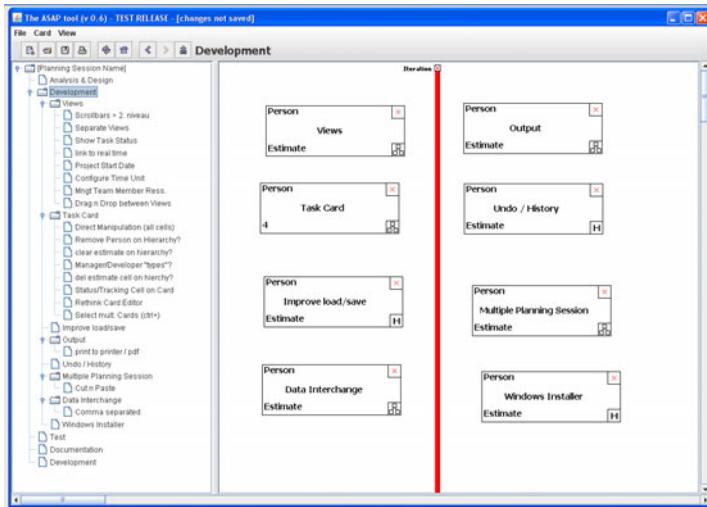
**Fig. 4.** Previous research on counterterrorism

In relation to *iMiner*, several collections of authenticated datasets of terrorist events that have occurred or were planned have been harvested from open source databases (i.e., TrackingTheTreat.com). Figure 5 shows the dataset on Al Qaeda.



**Fig. 5.** *iMiner* screenshot

Work has also been conducted on the ASAP tool (Figure 6) to assist software developers to perform structural analysis of software planning data [3]. Many of the spatial hypertext concepts and techniques that supports working with emergent and evolving structures [18] used in ASAP are domain independent and can be re-used in a tool that supports intelligence analysts working with terrorist information.



**Fig. 6.** ASAP screenshot

Finally, several prototypes have been constructed to harvest terrorist information from the Web: a focused web crawler for regular web pages [4], a tool to harvest information from RSS feeds [5], and a tool to harvest information from blogs [6].

## 4 Open Issues

As described in Section 3, we provide support for many of the processes based on novel models and advanced software tools. However, we have identified some open issues in relation to our work.

**Structure Analysis.** As mentioned, we have experiences from developing a structural analysis tool for the software planning domain. While some of the concepts from spatial hypertext can be re-used for the counterterrorism domain, it is still wide open how this should be done. Atzenbeck et al. [24] provide an analysis of the counterterrorism domain and lists requirements in relation to developing a structure analysis tool:

- Supporting the emergent and fragile nature of the created structure and fostering its communication among analysts.
- Integrating with the information sources used by the analyst, permitting them to be represented and structured in a common information space.
- Supporting awareness of, and notification based on, linked information across information source boundaries.
- Permitting multiple directions of thought through versioning support.

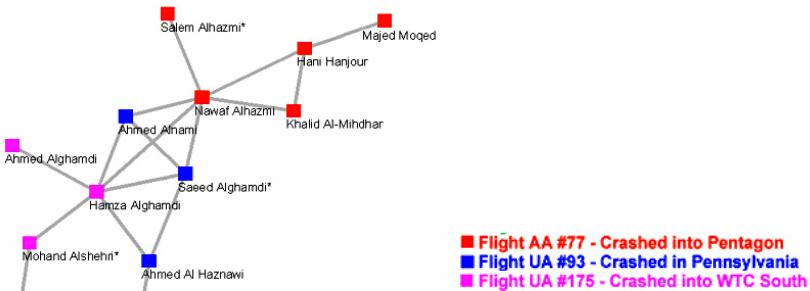
Thus, supporting emergent and evolving structure as a means for knowledge representation, communication, integration, versioning, awareness, and notification is central to this tool.

**Web Harvesting.** The three independent prototypes mentioned above form a good starting point for developing web harvesting tools that can support the data acquisition process in relation to the Web. The challenge is to combine the individual prototypes into an overall configurable, semi-automatic web harvesting tool. Related work regarding design and implementation of web crawlers [25], information gathering in a dynamic world [26], and studies of cyber communities in blogs [27] provides important pointers for this work.

**Knowledge base.** The knowledge base used by *iMiner* stores terrorist information in the form of triples:

*<subject, object, relationship>*

where “subject” and “object” are entities of interest and “relationship” is a link between exactly two entities [2]. This domain model with nodes (entities) and links (binary relations) supports development of advanced software tools to assist intelligence analysts. Figure 7 shows how this type of domain model can be used to model a complex terrorist networks (example from [28]).



**Fig. 7.** Part of 9/11 terrorist network [28]

Nodes are entities with attributes allowing relevant information to be stored about the entities. Social network analysis techniques can be used to identify key nodes in the network. This type of information can be used for network destabilization purposes. Taking out key nodes will decrease the ability of the network to function normally.

However, the above domain model also poses limitations. Links only exist as a text string describing the nature of the relation between two nodes (e.g., person A “met with” person B). Links are not first class entities with the same properties as nodes. This is in contrast to the fact that the links between the nodes provides at least as much relevant information about terrorist networks as the nodes themselves [29].

A domain model with links as first class entities (like nodes) will allow additional features to be built into the social network analysis and visualization tools:

- **Using Links Weights.** Currently, all links have the weight “1”. Having links as first class entities allows individual weights to be added to links. Weights can be based on information such as the reliability of the information and the level of the relation. Thus, links can be treated differently based on weights allowing more accurate information to be deducted from the terrorist network.

- **Finding Missing Links.** Investigative data mining techniques [30] could be used to suggest (predict) missing links in the terrorist network revealing relations that were previously unknown to the intelligence analysts.
- **Identifying Key Links.** Just like social network analysis techniques can be used to identify key nodes, they can also be used to identify key links in the terrorist network. A key link could for instance be “the flow of finances” between two persons. Taking out key links can also be used to destabilize terrorist networks.

These are just a few examples of how a more powerful domain model inspired by the basic hypertext node link model [17] can provide additional features for intelligence analysts. Future research is likely to reveal many additional features made possible by the new domain model.

## 5 Ongoing and Related Work

The above open issues are currently being addressed in various projects to further strengthen our toolbox approach to counterterrorism. Our goal is to provide a number of desktop tools that are grouped into three overall packages each containing a number of services relevant to counterterrorism. These services are designed and implemented in a way that enables them to interoperate and exchange information. Our current research on CrimeFighter can be divided into four areas:

1. **CrimeFighter Explorer** is a software package with various services aimed at acquiring data from open sources and extracting valuable information from the data. Hence, this package supports the data acquisition and information processing processes described above.
2. **CrimeFighter Investigator** is a software package that provides various services that enable an intelligence analyst to work with emergent and evolving structure of terrorist networks to uncover new relationships between people, places, events, etc. Hence, this package supports the analyzing knowledge process described above.
3. **CrimeFighter Assistant** is a software package with services that support analysis and visualization of terrorist networks. Terrorist network analysis is aimed at finding new patterns and gaining a deeper knowledge about terrorist networks. Terrorist network visualization deals with the complex task of visualizing the structure of terrorist networks. Hence, this package supports the interpreting and visualizing knowledge processes described above.
4. **CrimeFighter Toolbox Architecture.** In order for the developed tools and services to be able to interoperate and exchange information, overall software architecture for the toolbox is currently being developed to enable a service in one package to use a service in another package. For instance, the structure generated by the services of the CrimeFighter Investigator must be able to use the analysis and visualization services available in the CrimeFighter Assistant. Hence, a common interchange format for network structures has been defined to support the above scenario.

Figure 8 shows a screenshot from the first version of the CrimeFighter Assistant visualizing relevant analysis results in relation to the 2002 Bali Night Club Bombing

terrorist network. The visualization to the left highlights the 10 most important nodes and the 10 most important links in the terrorist network according to various terrorist network analysis measures.

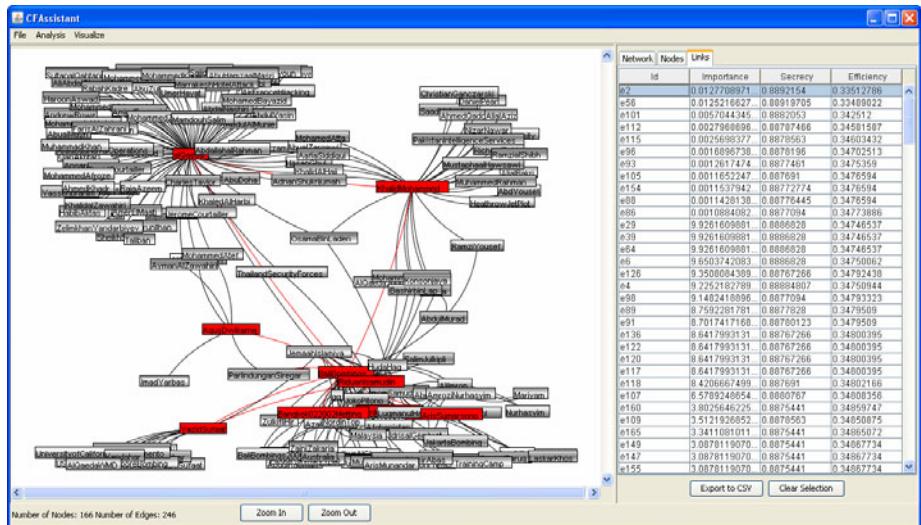


Fig. 8. Screenshot from CrimeFighter Assistant

The CrimeFighter approach towards a toolbox for counterterrorism is inter-disciplinary involving many different research topics as described in the previous sections. To our knowledge, no other approach provides a similar comprehensive coverage of tools and techniques to support the involved knowledge management processes.

The individual tools are based on theory from various research fields. Theory and related work from these fields are discussed throughout the paper – especially in Section 2.3 on techniques.

## 6 Conclusions

This paper described the latest research on the CrimeFighter toolbox for counterterrorism. The work reported in this paper has primarily made the following contributions:

- We have identified and described knowledge management processes, tools, and techniques that are central to the counterterrorism domain.
- We have developed and implemented advanced mathematical models and software tools that help automate and support knowledge processes for counterterrorism to assist intelligence analysts in their work.
- We have presented past, ongoing, and future work on CrimeFighter – a novel toolbox for counterterrorism that provides advanced support for the counterterrorism domain.

So far our tools and techniques have been tested with open source data from authenticated terrorist databases and the Web. As researchers, we do not have access to classified intelligence data. Our focus is on development of useful techniques and software tools. We hope to form formalized collaborations with intelligence agencies that wish to evaluate our tools on their classified data sets within their own secure settings and provide us with feedback on how well the tools work. Testing our tools and techniques with real intelligence data and real end users is the ultimate test to validate the value of our approach; this can take our research to the next level.

**Acknowledgements.** The authors wish to acknowledge the support from the Faculty of Engineering and the Maersk Mc-Kinney Moller Institute (both University of Southern Denmark) to establish the Counterterrorism Research Lab. This paper is an extended version of a paper previously published at the International Conference on Knowledge Management and Information Sharing (KMIS 2009) [31].

## References

1. Chen, H., Reid, E., Sinai, J., Silke, A., Ganor, B. (eds.): *Terrorism Informatics. Knowledge Management and Data Mining for Homeland Security*. Springer, Heidelberg (2008)
2. Memon, N., Wiil, U.K., Alhajj, R., Atzenbeck, C., Harkiolakis, N.: Harvesting Covert Networks: The Case Study of the iMiner Database. Accepted for the International Journal of Networking and Virtual Organizations (IJNVO). Inderscience Publishers (2010)
3. Petersen, R.R., Wiil, U.K.: ASAP: A Planning Tool for Agile Software Development. In: Proc. of the ACM Hypertext Conference, pp. 27–32. ACM Press, New York (2008)
4. Henriksen, K., Sørensen, M.: Design and Implementation of a Focused Web Crawler for use in Web Harvesting for Counterterrorism Planning Purposes. Project report. University of Southern Denmark (2009)
5. Knudsen, M.: Dynamic Web Harvesting Using RSS Feeds. Project report. University of Southern Denmark (2009)
6. Dasho, E., Puszczevicz, R.: Tools and Techniques for Counterterrorism: Web Mining and Social Network Analysis in Blogs. Project report. University of Southern Denmark (2009)
7. Mena, J.: *Investigative Data Mining for Security and Criminal Detection*. Butterworth-Heinemann, Butterworths (2003)
8. Devlin, K., Lorden, G.: *The Numbers Behind NUMB3RS: Solving Crime with Mathematics*. Plume (2007)
9. Alam, M.B.: Perceptions of Japanese Students on Terrorism. *Strategic Analysis* 27(2), 279–291 (2003)
10. Carpenter, M.A., Stajkovic, A.D.: Social network theory and methods as tools for helping business confront global terrorism: Capturing the case and contingencies presented by dark social networks. *Corporate strategies under international terrorism and adversity*. Edward Elgar Publishing (2006)
11. Raine, N.E., Rossmo, D.K., Comber, S.C.: Geographic profiling applied to testing models of bumble-bee foraging. *J. R. Soc. Interface* 6, 307–319 (2009)
12. Rossmo, D.K., Velarde, L.: Geographic profiling analysis: principles, methods, and applications. In: *Crime Mapping Case Studies: Practice and Research*, pp. 35–43. Wiley, Chichester (2008)
13. Bennell, C., Corey, S.: Geographic profiling of terrorist attacks. In: *Criminal Profiling: International Theory, Research and Practice*, pp. 189–203. Humana Press, Totowa (2007)

14. Canter, D.V., Hammond, L.: Prioritizing burglars: comparing the effectiveness of geographic profiling methods. *Police Pract. Res.* 8, 371–384 (2007)
15. Maciejewski, R., Hafen, R., Rudolph, S., Tebbetts, G., Cleveland, W.S., Grannis, S.J., Ebert, D.S.: Generating Synthetic Syndromic-Surveillance Data for Evaluating Visual - Analytics Techniques. *IEEE Computer Graphics and Applications* 29(3), 18–28 (2009)
16. Yan, P., Chen, H., Zeng, D.: Syndromic Surveillance Systems: Public Health and Bio-defence. *Annual Review of Information Science and Technology (ARIST)* 41, 425–495 (2007)
17. Engelbart, D.C.: Augmenting human intellect: A conceptual framework, Summary Report AFOSR-3233, Standford Research Institute (1962)
18. Shipman, F.M., Hsieh, H., Maloor, P., Moore, J.M.: The Visual Knowledge Builder: A Second Generation Spatial Hypertext. In: Proc. of the ACM Hypertext Conference, pp. 113–122. ACM Press, New York (2001)
19. Thomas, J., Cook, K.: A Visual Analytics Agenda. *IEEE Computer Graphics and Applications* 26(1), 10–13 (2006)
20. Heer, J., Boyd, D.: Vizster: Visualizing Online Social Networks. In: Proc. of the IEEE Symposium on Information Visualization (InfoVis 2005) (2005)
21. Adam, P., Shneiderman, B.: Balancing Systematic and Flexible Exploration of Social Networks. *IEEE Transactions on Visualization and Computer Graphics* 12(5), 693–700 (2006)
22. Xu, J., Chen, H.: CrimeNet Explorer: A Framework for Criminal Network Knowledge Discovery. *ACM Transactions on Information Systems* 23(2), 201–226 (2005)
23. Stasko, J., Gorg, C., Liu, Z., Singhal, K.: Jigsaw: Supporting Investigative Analysis through Interactive Visualization. In: Proc. of the IEEE Symposium on Visual Analytics Science and Technology, pp. 131–138 (2007)
24. Atzenbeck, C., Hicks, D.L., Memon, N.: Supporting Reasoning and Communication for Intelligence Officers. Accepted for the International Journal of Networking and Virtual Organizations (IJNVO). Inderscience Publishers (2010)
25. Shkapenyuk, V., Suel, T.: Design and Implementation of a High-Performance Distributed Web Crawler. In: Proceedings of 18th International Conference on Data Engineering, San Jose, CA, February, pp. 357–368. IEEE Computer Society, Los Alamitos (February 2002)
26. Hornung, T., Simon, K., Lausen, G.: Information Gathering in a Dynamic World. In: Alferes, J.J., Bailey, J., May, W., Schwertel, U. (eds.) PPSWR 2006. LNCS, vol. 4187, pp. 237–241. Springer, Heidelberg (2006)
27. Chau, M., Xu, J.: Using Web Mining and Social Network Analysis to Study the Emergence of Cyber Communities in Blogs. In: Terrorism Informatics, Knowledge Management and Data Mining for Homeland Security, pp. 473–494. Springer, Heidelberg (2008)
28. Krebs, V.: Mapping networks of terrorist cells. *Connections* 24, 45–52 (2002)
29. Gloor, P.A., Zhao, Y.: Analyzing Actors and Their Discussion Topics by Semantic Social Network Analysis. In: Information Visualization. IV, pp. 130–135 (2006)
30. Memon, N.: Investigative Data Mining: Mathematical Models of Analyzing, Visualizing and Destabilizing Terrorist Networks. Ph.D. Dissertation, Aalborg University, Denmark (2007)
31. Wiil, U.K.: Memon, Nasrullah, and Gniadek, Jolanta. 2009. Knowledge Management Processes, Tools and Techniques for Counterterrorism. In: Proceedings of the International Conference on Knowledge Management and Information Sharing (KMIS 2009), Funchal, Portugal, pp. 29–36. INSTICC Press (October 2009)

# Functional Analysis of Enterprise 2.0 Tools: A Services Catalog

Thomas Büchner, Florian Matthes, and Christian Neubert

Technische Universität München, Institute for Informatics  
Boltzmannstr. 3, 85748 Garching, Germany  
`{buechner, matthes, neubert}@in.tum.de`  
`http://wwwmatthes.in.tum.de`

**Abstract.** In recent years a new class of integrated web-based enterprise tools emerged facilitating team collaboration and knowledge management. In this paper we provide a detailed analysis of their concepts and services. We examined the following commercial and open source Enterprise 2.0 tools in detail: *Alfresco Share*, *Atlassian Confluence*, *GroupSwim*, *Jive SBS*, *Liferay Social Office*, *Microsoft Office SharePoint Server*, *Socialtext*, *Tricia*. Thereby, we derived an unifying multi-dimensional classification and evaluation framework. For each dimension we identified several technical criteria to characterize the functional capabilities of a given tool. Based on this schema we conduct a detailed evaluation for each particular tool. This work contributes to a better technical understanding of this emerging family of enterprise applications, highlights strengths and weaknesses of existing tools and identifies areas for further system research and development.

**Keywords:** Enterprise 2.0 software, Social software, Web-based collaboration, Knowledge management systems.

## 1 Motivation

In the last years a new class of collaboration tools emerged, which use so-called Web 2.0 technologies [8] to foster team collaboration and knowledge exchange. Since the objective of these tools is to adopt technologies and services proven successful on the Internet within enterprises, these are called Enterprise 2.0 tools [3,7]. As of today, there is a large number of applications in this category [4]. Those are complex integrated web-based tools, which offer a broad range of Web 2.0 concepts, like wikis, blogs, calendar, file share, search, and tagging.

An organization that wants to move towards ‘Enterprise 2.0’ is left the difficult decision which tool to choose. So far little guidance on how to classify and evaluate those tools exists. Comparing Enterprise 2.0 tools remains a challenging task because of the following reasons:

1. The tools differ greatly in the content types they support. On the one hand, there are simple tools, which concentrate on few concepts (e.g. wikis, files). On the other hand, there are applications, which offer a broad range of content types (e.g. calendar, tasks, issues, news). Since the only description of the tools available is in the

form of natural language marketing whitepapers, one has to dive deeply into those descriptions to identify the differences.

2. There is no agreed upon description of services an Enterprise 2.0 tool has to deliver. In [7] the following core services are identified (SLATES): search, links, authoring, tags, extensions, signals. Unfortunately, these terms are fuzzy and not used by all tools the same way. Since there is no uniform and detailed catalog of services available, comparing tools is difficult.

These difficulties and the observation, that there is a growing market for those tools [10] are the starting point for our work. The goal of this paper is to provide a detailed analysis of the concepts and services offered by existing Enterprise 2.0 tools based on a unifying multi-dimensional classification and evaluation framework.

In a first step, we had to choose, which applications to include in our initial analysis. The goal was to evaluate a representative set of relevant tools. As a first indicator we had a look at the Gartner magic quadrant in [4]. Since 2007 some new tools emerged, which we had to take into account. We focused our selection on big players, and additionally included Tricia<sup>1</sup>, a tool developed by members of our group.

Finally, we decided to evaluate the following applications (in alphabetical order): Alfresco Share<sup>2</sup>, Atlassian Confluence<sup>3</sup>, GroupSwim<sup>4</sup>, Jive SBS<sup>5</sup>, Liferay Social Office<sup>6</sup>, Microsoft Office Sharepoint Server<sup>7</sup>, Socialtext<sup>8</sup>, Tricia.

Due to space limitations, it is not possible to include all detailed results of our analysis in this paper. We will focus in the following on presenting our methodology as well as the catalog of services we created. The complete results can be found online at [2]. The online resource is intended to be expanded by additional tools in the future.

This paper is organized as follows: Section 2 gives an overview of related work. We then elaborate in Section 3 on how we analyzed the content types supported by each tool. In the Sections 4 and 5 we introduce a catalog of services, which we used to evaluate Enterprise 2.0 tools. In Section 6, we present the methodology of how we evaluated the given tools against the catalog. The paper concludes with a summary and an outlook.

## 2 Related Work

As shown in [6], Enterprise 2.0 tools are in the long-standing tradition of groupware and CSCW applications. In [9], a comparison of six commercial and academic CSCW systems is presented.

As already mentioned, [4] classifies 25 tools using alongside the non-functional dimensions *ability to execute* and *completeness of vision*. As a result, each tool falls into

---

<sup>1</sup> <http://www.infoasset.de>

<sup>2</sup> <http://www.alfresco.com/products/collaboration>

<sup>3</sup> <http://www.atlassian.com/software/confluence>

<sup>4</sup> <http://groupswim.com/products/collaboration-software>

<sup>5</sup> <http://www.jivesoftware.com/products>

<sup>6</sup> [http://www.liferay.com/web/guest/products/social\\_office](http://www.liferay.com/web/guest/products/social_office)

<sup>7</sup> <http://www.microsoft.com/Sharepoint/default.mspx>

<sup>8</sup> <http://www.socialtext.com>

one of the quadrants *challengers*, *leaders*, *niche players*, and *visionaries*. Two tools are classified as niche players, two applications come out as visionaries, and the great majority of tools has been classified as challengers.

There are some publicly available tool comparisons, which focus on tools for specific functionalities: WikiMatrix<sup>9</sup>, ForumMatrix<sup>10</sup>, Blog Comparison Chart<sup>11</sup>. These comparisons focus on one particular content type (wiki, forum, and blog).

Furthermore, there is work towards identifying services, Enterprise 2.0 tools should provide. In [7] the following services according the SLATES acronym are identified:

1. **Search** is required to find content objects,
2. **Links** connect and relate content objects,
3. **Authoring** makes it easy to contribute new content,
4. **Tags** form a bottom-up categorization system,
5. **Extensions** can be used to automatically compute recommendations,
6. **Signals** create awareness for the activities of other user.

In [5], an extension of SLATES is proposed, which in addition puts emphasis on the *social*, *emergent*, *freeform*, and *network-oriented* aspects. Nevertheless, as already mentioned in Section 1, these service descriptions are quite fuzzy and cannot be used to compare concrete Enterprise 2.0 tools in an objective manner.

### 3 Content Types

From a technical point of view an Enterprise 2.0 tool provides collaboration and communication services by many of *content objects*, e.g. wiki pages, blog posts, comments, files. Each application comes with a set of predefined *content types*, which realize the concepts provided by the tool. To get an overview of the capabilities of a given tool, it is helpful to first understand the supported content types and their associations.

As a first step in our survey, we therefore identified the core content types of each investigated tool and modeled them using a UML class diagram per application.

As it turned out, it is useful to differentiate between *core* content types, and *orthogonal* content types, which are needed to implement the services described in Section 5. Examples of orthogonal content types are *rating*, *tag*, *version*. To keep the models clean and simple, orthogonal content types are not modeled in our class diagrams, but rather discussed in Section 5. In the following, we will use the shorter term content type to mean core content type.

Due to space limitations, we cannot present the models of all surveyed applications here. As an example, the model of the content types provided by GroupSwim is shown in Figure 1. The models of all analyzed tools can be found online at [2].

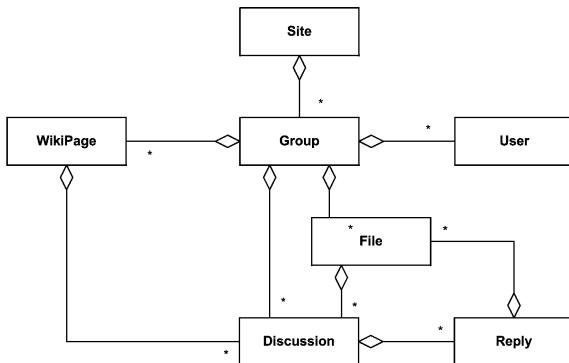
Different tools use different terminologies for conceptually similar content types. In our models, we use the terminology introduced by the given tool.

---

<sup>9</sup> <http://www.wikimatrix.org>

<sup>10</sup> <http://www.forummatrix.org>

<sup>11</sup> [http://www.ojr.org/ojr/images/blog\\_software\\_comparison.cfm](http://www.ojr.org/ojr/images/blog_software_comparison.cfm)

**Fig. 1.** Groupswim

## 4 Towards a Services Catalog

An Enterprise 2.0 tool provides for all of its content types *services* to make the content objects accessible. In the following we describe, how we created a services catalog, which can be used to compare and relate these tools. The basic idea of our approach is to analyze existing tools and to capture existing implemented services.

To narrow this task down, we only consider functionality provided out-of-the box by the main distribution of each tool. Several applications (e.g. Atlassian Confluence, Microsoft Office Sharepoint 2007) are complex extensible platforms and provide extensibility via a plugin mechanism or open APIs for third-party extension. These enhancements are not considered in our study.

As a second restriction, we only consider services, which are visible to the end-user. Therefore, maintenance and configuration services are not part of our services catalog.

Furthermore, we focus on a functional analysis. Non-functional aspects, such as e.g. cost, extensibility, performance, deployment type, ease of implementation, etc., are not regarded. These dimensions could be additionally included in a later version of our schema.

Initially, we gathered all available services of the investigated tools. Indeed, most of the applications support similar services, but the terminology used often varies, e.g. the creation of tags vs. the assignment of labels. Therefore, we consolidated these similar concepts to a general service description and extracted short service names, e.g.:

*Private Tags*: The usage of *private* tags is supported. Private tags are only visible to the creator and not to other user of the tool.

This representation of the service short name (*italic*) followed by the general service description is used in the services catalog presented in Section 5.

In some rare cases we extended the service description to a more complete and more reasonable specification from a technical point of view. For example, Microsoft Office SharePoint 2007 gives access to the *title* property of an MS Office document. Adapted from that, we inferred the more general service description: Access and manipulation

Service Context	Service Category	Service	Vendor							
			Alfres-co	Confluence	Group-Sync	Jive SBS	MOSS	Social-text	Tricia	Liferay
<b>Content-Centric</b>	<b>Authoring</b>	WYSIWYG-Editor	●	●	●	●	●	●	●	●
		Support for tables, images, and media objects	●	●	●	●	●	●	●	●
		Input support for link creation	○	●	●	●	○	●	●	●
		Autosave	●	●	●	●	○	●	●	○
		Description of all content objects by rich markup text	●	●	●	○	●	●	●	●
		Spell checking	○	○	●	●	●	○	○	●
		Concurrent Editing	○	○	○	●	○	●	○	○
		Offline Editing	○	○	○	○	○	●	○	○
	<b>Link management</b>	Human-readable permalinks for all content objects	●	●	●	●	●	●	●	●
		Stable URIs for containers and actions	●	●	●	●	●	●	●	●
		Labeling of invalid links	○	●	●	●	○	●	●	○
		Search for invalid links	○	○	○	○	○	●	●	○
	<b>Tagging</b>	Automatic propagation of link updates	○	●	●	●	○	●	●	○
		Tag support for all content objects	●	●	●	●	○	●	●	●
		Input support for tag creation	●	●	●	●	●	●	●	●
		Tag usage overview	●	●	●	●	○	●	●	○
	<b>Search</b>	Private Tags	○	●	○	○	○	○	○	○
		Full-text search over all content	●	●	●	●	●	●	●	●
		Search content of files	●	●	●	●	○	●	●	●
		Highlighting of search hits	○	●	●	●	●	●	●	○
		Advanced search operators	●	●	●	○	●	●	●	●
		Sorting	○	●	●	●	●	●	●	●
		Filtering	○	●	●	●	●	●	●	●
	<b>Version management</b>	Safety net through content revisions and audit trail	●	●	●	●	●	●	●	●
		Annotation and classification of revisions	○	●	●	●	●	●	●	●
		Human readable presentation of revision differences	○	●	●	●	●	●	●	●
		Restore	●	●	●	●	●	●	●	●
		Access control for versions	○	○	○	○	○	○	●	○
	<b>Desktop integration</b>	Undelete	○	○	○	○	○	○	○	●
		File access	●	●	○	○	●	○	●	●
		Metadata	○	○	○	○	●	○	○	○

**Fig. 2.** Ratings Content-Centric

of all file metadata, e.g. title, description, author, etc. Based on this generalized service description, we evaluated the implementation of these services for all given tools. Our methodology for this evaluation is presented in Section 6. Overall, we derived 49 Enterprise 2.0 core services.

Since some of the inferred services are similar to each other, we arranged them into 13 more general categories. For instance, the category ‘Link Management’ contains services dealing with the handling of references (links) between content objects.

Based on the identified 13 service categories, we determined two reasonable services not supported by any tool at all. These services are relevant from our point of view, hence we decided to exclude them from the core services catalog. Nonetheless, these services are described in Section 5.4.

We observed, that the context of a given service is either focused on content objects, or on aspects concerning the user of a tool. We therefore classified the 13 categories in *content-centric* (cf. Figure 2) and *user-centric* (cf. Figure 3). Nevertheless, a few services cannot be assigned to exactly one of these classes. Those services are part of a third class *orthogonal* (cf. Figure 3), called orthogonal services.

Service Context	Service Category	Service	Vendor							
			Alfres-co	Confluence	Group-Sync	Jive SBS	MOSS	Social-text	Tricia	Liferay
User-Centric	Access control	Creation of groups and invitation of new members by users	●	○	●	●	●	●	●	*
		Uniform, flexible, and fine granular access control concept for all content types	●	●	○	●	●	●	●	*
		Functional groups for access control	○	●	○	○	●	○	●	*
		Content of any type may be made available for anonymous users	○	●	●	○	○	●	●	*
		Smooth transition between the usage modes not logged on and logged on	○	●	○	○	●	●	●	*
	Feedback	Spam avoidance	○	○	●	○	○	○	○	*
		Comments to content of any type	●	●	●	●	●	●	●	●
		User ratings	○	○	●	○	○	○	○	●
	Social Networking	Anonym post of comments	○	●	○	○	○	●	●	*
		Support for social network building	○	○	○	●	●	○	○	○
Orthogonal	Awareness	Fine granular access control for user profile properties	○	○	○	○	●	○	○	○
		Tracking of other users' activities	○	○	●	●	○	●	○	○
		Tracking of activities on content and container objects	●	●	●	●	●	●	○	●
	Usage Analytics	Support for different message channels	●	●	●	●	●	●	○	●
		Usage statistics down to the level of individual content items	○	●	○	○	●	○	○	○
		Search words statistics	○	○	○	○	●	○	○	○
		Consistent GUI	●	●	●	●	●	●	●	●
	Personalization	Adaptable look&feel for certain functional areas	○	●	○	○	●	○	●	○

**Fig. 3.** Ratings User-Centric and Orthogonal

## 5 Services Catalog

A service description, a classification, and a service context constitute the dimensions of our services catalog. The following section introduces the catalog in detail.

### 5.1 Content-Centric Services

**Authoring.** A significant Enterprise 2.0 tool characteristic is the collaborative web-based creation and manipulation of content respectively content objects. We categorize all services dealing with this process as ‘Authoring’.

**WYSIWYG-Editor:** The content creation process is assisted by a hypertext editor. The editor enables users to create plain text and additionally provides functions to enrich this content with markup (e.g. HTML, wiki markup) for layouting purpose. We expect the editor to be a WYSIWYG-Editor (What-You-See-Is-What-You-Get), i.e. changes on the contents’ layout are immediately visible for the user. The editor enforces a strict separation of content and layout. Nevertheless, power users sometimes prefer being able to edit the underlying markup manually. For this reason, an advanced view is provided to enable modifications of the markup language directly. If HTML is used as the underlying markup language, the system has to take measures to prevent Cross-Site-Scripting (XSS) attacks. Finally, sections from Microsoft Office documents can be pasted into the editor, thereby transforming the original layout to the corresponding markup language (as far as this is possible).

*Support for tables, images, and media objects:* Beside text, tables, images, and rich media objects (video, flash, and mp3 objects) can be embedded using the editor.

*Input support for link creation:* To reference other content objects or container objects links can be defined. The WYSIWYG-Editor assists the creation of valid links to all existing types by giving suggestions.

*Autosave:* When editing hypertext, an autosave functionality automatically creates server-side backups to prevent changes get lost in case of a broken Internet connection. Moreover, if the user leaves a page with pending changes without saving the changes, a corresponding warning message is shown.

*Description of all content objects by rich markup text:* In contrast to ‘WYSIWYG-Editor’, where the requirement is the general existence of a WYSIWYG-Editor, we claim here, that all content objects can be described using hypertext in the exact same manner. Additionally, the WYSIWYG-Editor provides a set of predefined styles for layouting purpose.

*Spell checking:* To increase the contents’ quality, the editor provides spell checking functionality.

*Concurrent Editing:* To prevent concurrent conflicting edits, the system gives a warning message, if a user starts editing a page, which is currently being edited by someone else.

*Offline Editing:* Even if no Internet connection is available, all content objects can be modified offline. In this case the edits are stored locally on the client machine. When going online the objects are synchronized with the backend. The editing experience in the on- and offline mode should be as close as possible.

**Link Management.** Link management are services dealing with the handling of references to content (e.g. wiki pages, files) and container objects (e.g. wikis, directories).

*Human-readable permalinks for all content objects:* All content objects are referenced by stable, human-readable URLs, so called *permalinks*.

*Stable URLs for containers and actions:* Container objects, collections of objects, and actions are referenced by stable URLs. Collections are e.g. last modified wiki pages, blog posts by user xyz.

*Labeling of invalid links:* The system recognizes and highlights invalid links. This is visible in the WYSIWYG-Editor.

*Search for invalid links:* To detect invalid links, the system provides a search mechanism. This helps keeping the system clean of broken links.

*Automatic propagation of link updates:* If the URL of a content object changes (e.g. by renaming a wiki page or a file), this change is propagated and all affected links are adapted to the new URL. Links to deleted objects are highlighted automatically as being invalid.

**Tagging.** Tagging constitutes the process of collaboratively building a bottom-up categorization system. This subsection considers tagging services for content objects.

*Tag support for all content objects:* Multiple tags can be assigned to all content objects. The only exception concerns the tagging of persons. We do not expect this service be available to prevent misuse.

*Input support for tag creation:* The system supports the creation of tags by showing existing tags and their usage frequency (e.g. by font size or number).

*Tag usage overview:* An overview of all existing tags shows the usage frequency numerically and visually as a tag cloud.

*Private Tags:* The usage of *private* tags is supported. Private tags are only visible to the creator.

**Search.** This category subsumes services regarding finding content.

*Full-text search over all content:* A unified text search over all content objects exists. Comments, tags, and attributes of the content objects are included in the search as well.

*Search content of files:* The full textual content of files is searched.

*Highlighting of search hits:* Occurrences of the search terms are highlighted in the search results using a clear representation.

*Advanced search operators:* The text search features AND, OR, and NOT operators, wildcards, and search for phrases are supported.

*Sorting:* The default sorting of the search results is by relevance. Additionally, it is possible to sort by last modification date and by last modifier.

*Filtering:* The search results can be filtered by content type, tags, modification date, and modifier.

**Version Management.** The category Version Management contains services concerning tracing the evolution of the content objects within their life-cycle.

*Safety net through content revisions and audit trail:* For wiki pages and files a version history is maintained, which includes information about modifier and modification date.

*Annotation and classification of revisions:* The modifier may provide a version comment for each change. It is possible to categorize changes according to their importance.

*Human readable presentation of revision differences:* The system highlights differences between versions in a clear and understandable way.

*Restore:* It is possible to restore old versions.

*Access control for versions:* The version management takes access control settings into account: versions adopt their access control setting when they are created and enforce this setting later on.

*Undelete:* It is possible to restore even deleted wiki pages and files. This also recovers the complete version history.

**Desktop File Integration.** Desktop file integration is about services dealing with the direct and flexible access to files stored in the Enterprise 2.0 tool.

*File Access:* Additionally to web access, files can be accessed using standardized protocols, like SMB, WebDAV, and FTP.

*Metadata:* Embedded file metadata (e.g. in Word, PDF, JPG documents) is adopted and can be accessed and manipulated.

## 5.2 User-Centric Services

**Access Control.** Services dealing with authorization management for content objects are part of this category.

*Creation of groups and invitation of new members by users:* Users can create new user profiles and user groups and invite new members according to given membership policies.

*Uniform, flexible, and fine granular access control concept for all content types:* A uniform, flexible and fine granular access control concept exists. This is uniform and consistent for all object types.

*Functional groups for access control:* Functional groups are used for definition of access rights (cf. ‘Uniform, flexible, and fine granular access control concept for all content types’). During the assignment of functional groups input support is provided.

*Content of any type may be made available for anonymous users:* It is possible to make content of any type available for known as well as for anonymous users.

*Smooth transition between the usage modes not logged on and logged on:* The system provides a smooth transition between the usage modes not logged on and logged on. i.e. the primary requested resource (e.g. page) is accessed after successful login.

*Spam avoidance:* The system provides mechanisms to prevent spam attacks. Captchas (visual and audio) are used for all objects anonymous users can contribute to. This feature is not relevant, if anonymous user are not supported at all.

**Feedback.** Feedback considers services for the management and exchange of opinions.

*Comments to content of any type:* Users can write comments to content of any type. The creation of comments can be disabled.

*User ratings:* It is possible to rate the quality of any content object. This can be disabled.

*Anonymous post of comments:* Anonymous user may post comments to content of any type. This feature is not relevant if anonymous user are not supported at all.

**Social Networking.** This category is dealing with services about the informal aggregation of user groups.

*Support for social network building:* Users can build up a social network, i.e. they can set them in relation to each other by inviting other users to be a ‘friend’, ‘colleague’. The invitation can be accepted or rejected by the invitee.

*Fine granular access control for user profile properties:* Every user may provide a profile page with personal information. Parts of the profile (e.g. sensitive attribute of the user) page can be protected against objectionable access.

**Awareness.** Awareness subsumes services about tracking system activities.

*Tracking of other users’ activities:* Users can track the activities of others users or user groups.

*Tracking of activities on content and container objects:* Users can track the activities on content and container objects.

*Support for different message channels:* Users can configure different channels for receiving messages for tracked activities. These channels are: dashboard, RSS, and e-mail.

**Usage Analytics.** All services dealing with statistical analysis are included in this category.

*Usage statistics down to the level of individual content items:* The system provides statistics for the usage of content. Thus, it can be evaluated how many users accessed a certain content object, the frequency of access and the access point of time.

*Search words statistics:* The system provides statistics, which search words led to the site.

### 5.3 Orthogonal Services

**Consistent Graphical user Interface.** This category regards usability services and handling of the graphical user interface.

*Consistent presentation of actions and views:* The graphical user interface is consistent and clearly structured. For all object types the presentation of actions and views is uniform.

**Personalization.** Personalization comprises services dealing with the adaptivity of the system according user needs.

*Adaptable look&feel for certain functional areas:* The user can customize certain functional areas of the graphical user interface. Additionally, an existing corporate design can be integrated overall.

### 5.4 Additional Services

**Usage Analytics.**

*Referer statistics:* The system keeps track of pages the accessing users came from.

**Feedback.**

*Searchable and sortable ratings:* User ratings can be used as filter and sorting criteria in the unified search.

## 6 Rating Methodology

Based on the introduced services catalog, we performed an evaluation of eight Enterprise 2.0 tools. In this process, we evaluated the capabilities of all tools with regard to all of our services. Thereby we applied ratings between 0 and 4, 0 stands for no capabilities, 4 stands for complete coverage of the service. In case a service is only partially covered by a tool (i.e. a rating between 1 and 3), we provide a detailed explanation of

what exactly is missing (cf. Figure 4). These explanations are available at [2]. We do not comment on services having full capabilities as well as those achieving no score at all.

As described in Section 5, some service descriptions are more general than the capabilities of all tools. This implies for some services, that no tool obtains the full score, e.g. for service ‘Metadata’ in the category ‘Desktop File Integration’.

Service Context	Service Category	Service	Vendor							
			Alfres- co	Con- fluence	Group- Swim	Jive- SBS	MOSS	Social- text	Tricia	Liferay
Content-Centric	Authoring	WYSIWYG-Editor	█	█	█	█	█	█	█	█
- wiki-syntax and HTML supported, but the conversion from wiki-syntax to HTML (HTML to wiki-syntax) is not supported - no prevention of XSS attacks										

**Fig. 4.** Ratings for WYSIWYG-Editor, Authoring

In the following, we give an example of a concrete service evaluation. In the sample we consider the core service ‘WYSIWYG-Editor’ within the category ‘Authoring’ (cf. Figure 4). Jive SBS, Socialtext, and Tricia have full capabilities, so they get a full rating and no explanations are necessary. The tools Alfresco, GroupSwim, Microsoft SharePoint, and Confluence do not support *paste sections from MS Office documents*, so pasting from these document types either removes all formatting information or in some cases inserts unwanted style information into the target content. Additionally, no manual markup editor for power users is provided by Alfresco, as demanded by the service description. The WYSIWYG-Editor used in Liferay supports wiki-markup as well as HTML. Unfortunately, the conversion from wiki-markup to HTML and vice versa is not supported, so when changing the representation, markup information is lost. Furthermore, the manual HTML markup editor does not prevent XSS attacks. The resulting ratings are visualized in table 4. The ratings (0-4) are presented in a visual pie chart representation.

We did not calculate a total rating for each service category, because this would imply to define weightings for all service ratings. The decision of how important a particular service is, remains to the user of the evaluation framework.

For several reasons we cannot obtain a rating in some cases, e.g. caused by the occurrence of errors in the test scenario. This services are marked with a \* character (cf. table 3).

The complete analysis with all additional explanations can be accessed online at [2].

## 7 Conclusions and Outlook

There is a growing market for Enterprise 2.0 tools and it is difficult to compare existing tools against each other. Our paper on the one hand increases the transparency of this market by providing a methodology for comparing given tools. On the other hand, we applied this methodology and actually compared eight relevant tools.

We see potential to improve our existing methodology and comparison in the following points:

1. To broaden our analysis we will analyze more tools. Specifically, we want to analyze the IBM Lotus tools family <sup>12</sup>.
2. To improve our analysis we are in the process of getting feedback from the tool vendors. This feedback will improve our services catalog as well as the actual ratings for the tools.
3. An interesting extension of our comparison would be to also incorporate non-functional criteria, such as e.g. deployment options, performance, scalability.

In order to increase the ratings transparency we will make our test scenarios publicly available. Thereby, the independent evaluation of tools by means of the services catalog is facilitated for companies which want to move towards Enterprise 2.0. Furthermore, we will provide screenshots in case a service is only partially covered by a tool (i.e. a rating between 1 and 3) online at [2].

Since this survey was mainly conducted in the beginning of 2009, it will be interesting to watch, how current versions of the considered tools as well as new emerging tools in 2010 affect the stability of the services catalog and the particular ratings. Specifically, we will analyze the capabilities of Microsoft SharePoint 2010 <sup>13</sup> in order to capture the evolution of the services catalog and to compare the evaluation results with those of the MOSS 2007. It will also be interesting to examine, whether the services identified in Section 5.4, will be implemented in current tools.

Furthermore, based on the identified services and categories, it could be lucrative to conduct empirical studies on how effective their actual use is. Thereby, the experience gained from industry cases [1] could be included.

## References

1. Back, A., Koch, M., Smolnik, S., Tochtermann, K. (2010),  
<http://www.e20cases.org/lang/de/fallstudien>
2. Büchner, T., Matthes, F., Neubert, C.:  
<http://wwwmatthes.in.tum.de/wikis/enterprise-2-0-survey/home>
3. Bughin, J.: The Rise of Enterprise 2.0. Journal of Direct. Data and Digital Marketing Practice 9(3), 251 (2008)
4. Drakos, N.: Magic Quadrant for Team Collaboration and Social Software. Gartner Research, ID Number: G00151493 (2007)
5. Hinchcliffe, D.: The state of Enterprise 2.0 (2007),  
<http://blogs.zdnet.com/Hinchcliffe/?p=143>
6. Koch, M.: CSCW and Enterprise 2.0 - towards an integrated perspective. In: Proc. Conf. Bled eConference eCollaboration, pp. 416–427 (2008)
7. McAfee, A.: Enterprise 2.0: The Dawn of Emergent Collaboration. IEEE Engineering Management Review 34(3), 38–38 (2006)

---

<sup>12</sup> <http://www-01.ibm.com/software/de/lotus>

<sup>13</sup> <http://sharepoint2010.microsoft.com>

8. O'reilly, T.: What is web 2.0: Design patterns and business models for the next generation of software. Social Science Research Network Working Paper Series (August 2007)
9. Rama, J., Bishop, J.: A survey and comparison of CSCW Groupware applications. In: Proceedings of the 2006 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries, South African Institute for Computer Scientists and Information Technologists, Republic of South Africa, pp. 198–205 (2006)
10. Young, G.O.: Top Enterprise Web 2.0 Predictions for 2008, Forrester Report (2008)

# Scenario Process as a Community for Organizational Knowledge Creation and Sharing

Hannu Kivijärvi<sup>1</sup>, Kalle A. Piirainen<sup>2</sup>, and Markku Tuominen<sup>2</sup>

<sup>1</sup> Aalto University School of Economics, Department of Information Systems Science  
Runeberginkatu 22-24, 00101 Helsinki, Finland

<sup>2</sup> Lappeenranta University of Technology, Faculty of Technology Management  
Department of Industrial Management, Skinnarilankatu 34, 53850 Lappeenranta, Finland  
Hannu.Kivijarvi@aalto.fi  
{Kalle.Piirainen,Markku.Tuominen}@lut.fi

**Abstract.** Uncertainty and radical changes in the environment challenge decision makers in modern organizations. Knowledge of the past and present and also the insights from the future form the necessary conditions for successful decision making. Knowledge of the present can be used to create knowledge about the future through the scenario process. The paper presents two case studies, which support the proposition that the scenario process supports creation of knowledge in organizations. The scenario process can be a community and thus can offer not only artifacts of knowledge, but also a venue where decisions can be rehearsed and evaluated against plausible futures. We arrive to the conclusion that the scenario process can support knowledge sharing and creation, while it makes the assumptions of the organization explicit through the scenarios and also results in projection of plausible futures to come.

**Keywords:** Personal knowledge, Organizational knowledge, Communities of practice, Virtual communities, Scenarios, Scenario planning.

## 1 Introduction

Knowledge and knowledge sharing are important to any modern organization. The quality of decision making depends on creation, transformation and integration of knowledge across individuals and organizational groupings. Every decision situation in organizational decision making involves a decision maker(-s), desired outcomes or objectives and goals, at least two decision alternatives, and an environment or a context. In addition, an implicit assumption of every decision situation is the future oriented conception of time; decisions are meaningful only in reference to the future, not to the past or present.

The rapid rate of technological, economic and social changes that have an effect on organizational environment has increased the need for foresight. Because the future in absolute term is always at least partly unknown, it cannot be predicted exactly. The external environment is not under the control of the organization and therefore the environment is a source of uncertainty. The ability to see in advance is rooted in

present knowledge and in partially unchanging routines and processes within an organization. The quality of attempts to foresee is finally grounded on our knowledge and ability to understand the present position deeply enough.

A popular tool to develop foresight is the scenario process, in which the aim is to produce plots that tie together the driving forces and key actors of the organizational environment [30] in future scenarios. Although future oriented, scenarios are also projections of the known, extensions of the present situation over into the unknown future. However, even if scenarios are projections of the known they still serve to develop foresight and they have additional value as representations of organizational knowledge.

Concepts like the community of practice [21] and networks of practice [6] are used to explain the organizational conditions favoring knowledge creation and sharing and innovation. The most favorable contents of these arrangements certainly depend on factors such as the organizational context, the experiences and other capabilities of the members, and management style.

This paper discusses the theoretical basis for creating conditions to support formation of a community to enable knowledge sharing and goes on to propose such a condition or an artifact. We argue that the electronically mediated scenario process can act as a community and enable the participants to share their knowledge while exploring the future. The potential value of the proposed approach is evaluated mainly by epistemological criteria. Thus the question to which we seek answer is: How the scenario process can support organizations in their strive towards knowledge creation?

The remainder of the paper is organized as follows. The second section discusses knowledge and its creation in organizational contexts. The third section presents the scenario process and discusses its properties as a venue for knowledge creation. The fourth and last section discusses the results and presents conclusions at theoretical and practical levels.

## 2 Conceptual Background

### 2.1 Knowledge and Knowing

Knowledge is traditionally interpreted as a singular, independent object. Another, procedural interpretation of knowledge is to see it as a path of related steps [8]. Tsoukas and Vladimirov [33, p. 979] relate knowledge to a person's ability to draw distinctions: "Knowledge is the individual ability to draw distinctions, within a collective domain of action, based on an appreciation of context or theory, or both." According to this definition, a person who can draw finer distinctions is more knowledgeable. Making distinctions and judgments, classifying, structuring, placing order to chaos, are capabilities of an expert who has knowledge.

If decision making is not a synonym for management, as Simon [31] has argued, decision making is still undoubtedly at the core of all managerial functions. When a decision is made, the epistemic work has been done and the physical work to implement the decision can start. The value of knowledge and information is ultimately evaluated by the quality of the decisions made. Making decisions involves

also making distinctions, categorizations and judgments – we need to search for and structure alternatives. According to Emery [12, p. 67] information has value only if it changes our view of the world, if our decisions are sensitive to such a change, and if our utility is sensitive to difference in decisions. Thus, information is valued through decisions and because information and knowledge are relative, the same logic can be used to value knowledge, too. Kivijärvi [18] has elaborated the characterization of knowledge further and defines knowledge as the individual or organizational ability to make decisions; all actions are consequences of decisions. Also Jennex and Olfman [15, p. 53] note that “...decision making is the ultimate application of knowledge”.

When Polanyi [29] talks of knowledge in his later works, especially when discussing tacit knowledge, he actually refers to a process rather than objects. Consequently, we should pay more attention to tacit knowing rather than tacit knowledge. Zeleny [36] characterizes the relationship of explicit and tacit knowledge much in the same way as Polanyi. The procedural interpretation of knowledge is to see it as a path of related steps, embedded in the process of ‘knowing’, in the routines and actions that come naturally for a person who knows [8], [36]. Cook and Brown [11] also emphasize that knowing is an important aspect of all actions, and that knowledge will manifest itself during the knowing process.

Polanyi [28] tied personal dimension to all knowledge and his master-dichotomy between tacit and explicit knowledge has shaped practically all epistemological discussion in knowledge management field. According to Polanyi tacit knowledge has the two ingredients, subsidiary particulars and focal target (or proximal and distal [29, p. 10]). Subsidiary particulars are instrumental in the sense that they are not explicitly known by the knower during the knowing process and therefore they remain tacit. Thus, “we can know more than we can tell” [29, p. 4] or even “we can often know more than we can realise” [23, p. 114], which makes it easy to act but often challenging to articulate the principles behind the action.

To move from personal level to organizational level, Tsoukas and Vladimirov [33, p. 981] write “Organizational knowledge is the set of collective understanding embedded in a firm”. It is “the capability the members of an organization have developed to draw distinctions in the process of carrying out their work, in particular concrete contexts, by enacting sets of generalizations ... whose application depends on historically evolved collective understandings and experiences” [33, p. 983]. In other words, organizational knowledge can be seen as the capability the members of an organization have developed to make decisions in the process of carrying out their work in organizational contexts [18].

## **2.2 Contexts for Knowledge Creation and Sharing**

Lave and Wenger [21, p. 98] introduced the concept of community of practice and regarded it as “an intrinsic condition for the existence of knowledge”. Communities of practice have been identified as critical conditions for learning and innovation in organizations, and they are formed spontaneously by work communities without the constraints of formal organizations. According to Lesser and Everest [22, p. 41] “Communities of practice help foster an environment in which knowledge can be created and shared and, most importantly, used to improve effectiveness, efficiency and innovation”. In other words, a community of practice can form the shared

context, which supports the recipient decoding a received message with the same meaning the sender has coded it [14]. Although the communities develop informally and spontaneously, the spontaneity can be structured in some cases [7].

When people are working together in communities, knowledge sharing is a social process, where the members participate in communal learning at different levels and create a kind of ‘community knowledge’ following the procedural perspective to knowledge. According to the studies on communities of practice, new members learn from the older ones by being allowed to participate first in certain ‘peripheral’ tasks of the community and are approved to move to full participation later. After the original launching of the concept of community of practice, a number of attempts have been made to apply the concept to business organizations and managerial problems [5]. Recent studies on communities of practice have paid special attention to the manageability of the communities [32], alignment of different communities, and the role of virtual communities [17]. Gammelgaard and Ritter [14], for example, propagate virtual communities of practice, with certain reservations, for knowledge transfer in multinational companies.

To sum up, the general requirements for a community are a common interest, a strong shared context including own jargon, habits, routines, and informal ad hoc relations in problem solving and other communication [1]. An important facet of a community of practice is that the community is emergent, and is formed by individuals who are motivated to contribute by a common interest and sense of purpose. A cautious researcher might be inclined to use the term quasi-community or some similar expression in the case of artificial set-ups, but in the interest of being succinct, we use the word community in this paper also for non-emergent teams.

### 2.3 Scenarios and the Scenario Process

Kahn and Wiener [16, p. 33] define scenarios as “Hypothetical sequences of events constructed for the purpose of focusing attention to causal processes and decision points”, where the development of each situation is mapped step by step, and the decision options are considered along the way. The aim is to answer the questions “What kind of chain of events leads to a certain event or state?” and “How can each actor influence the chain of events at each time?” This definition has similar features as Carlile and Rebentisch’s [8] definition of knowledge as a series of steps as discussed above.

Schwartz [30] describes scenarios as plots that tie together the driving forces and key actors of the environment. In Schwartz’ view the story gives a meaning to the events, and helps the strategists to see the trend behind seemingly unconnected events or developments. The concept of ‘drivers of change’ is often used to describe forces such as influential interest groups, nations, large organizations and trends, which shape the operational environment of organizations [4], [30]. We interpret that the drivers create movement in the operational field, which can be reduced to a chain of related events. These chains of events are in turn labeled as scenarios, leading from the present status quo to the defined end state during the time span of the respective scenarios.

The scenario process is often considered as a means for learning or reinforcing learning, as discussed by Bergman [2], or a tool to enhance decision making

capability [9]. Chermack and van der Merwe [10] have proposed that often participation in the process of creating scenarios is valuable in its own right. Accordingly a major product in successful scenarios is a change in the participants view to the world and the subject area of the scenarios [10], [35]. They argue further that even the most important aim of scenario process is to challenge the participants' assumptions of the future and let them to re-examine their assumptions analytically. In other words, the learning process enables the participants to examine their assumptions and views, challenges them and as a result, will help changing mental models to suit the environment [35]. This is another feature that has echoes in knowledge management field, as Emery [12] proposed that one of the conditions information has to fulfill to have value, is that it changes our worldview, and here we can argue that participation in scenario process will potentially change the participants worldview.

When we contrast these properties of scenarios as a product and a process to the discussion about knowledge, we will notice that knowledge is manifested in knowing, decision making and action. Scenarios on the other hand enable simulation of action, through analysis of the current situation and analytical projections from the assumptions. So we can propose that scenarios can be viewed 1) as a process of (organizational) learning, but at the same time scenarios 2) as projections of future can be manifestations of knowledge about the past and present, and lastly scenarios 3) as stories of plausible futures can act as a rehearsal for the future, testing of present knowledge and routines in different environments.

## 2.4 Linking the Conceptual Elements

We proposed that in its deepest sense knowledge is and manifests as capability to make decisions. Scenarios, as discussed above, can be linked to organizational learning and knowledge on multiple levels. Firstly, the process forces the participants to think about the present, the drivers of the situation and where does it evolve, and the process guides the participants to critically examine their mental models through critical discussion in the group and to converge toward a commonly agreed statement of futures. Secondly, the scenarios as a product codify and make the assumptions explicit and illustrate the created knowledge of the future at that given point of time. And thirdly, when the group creates plausible stories of the world of tomorrow, they can be used as a framework for reflecting existing knowledge and mental models, and their fitness to the new situations.

Scenarios as artifacts are projections of present knowledge and aim to increase the organizational capability to make decisions. By definition they are a type of organizational knowledge but knowledge is also tied to action, and scenarios are a kind of 'quasi-action' where one can evaluate present knowledge and planned action against the scenarios. In addition to the scenario artifacts, the process of creating them helps the members of the community to use their subsidiary awareness of the future. All knowledge has a tacit, hidden dimension, which is only partly consciously known, whereas the other part is instrumental and is known only at the subsidiary level. Subsidiary awareness forms a background or context for considering the future. While it cannot be directly articulated in explicit form, the subsidiary knowledge will be manifested when those foresights are used in the knowing process. Thus we argue that the scenario process is a foreseeing process where the subsidiary awareness of each

participant is transformed into organizational scenarios. If we accept these premises, we can argue that scenarios enable ‘rehearsing for the future’ and presenting knowledge of the present as well as future.

The remaining question is then how to manage the process effectively to organize and transform available knowledge to logical scenarios. We propose that a community of practice would be a favorable condition to create scenarios. The experimental community we propose in this study is a group support system (GSS) facility, which is used to mediate the interaction and to support the community in the task of composing scenarios. One question is whether the process satisfies the conditions of being a community, and if the community in the case is not emergent, but purposefully set up, is still a community? The answer of Amin and Roberts [1] would most likely be ‘yes and no’, and the short-lived community this paper presents would be classified as a ‘creative community’, where the base of trust is professional and the purpose is to solve a problem together.

### 3 Knowledge Creation and Sharing in the Scenario Process

#### 3.1 Overview of the Scenario Process

The discussion above presented the argument that scenarios can enable knowledge creation and storing it. The method adopted in this study is the intuitive decision-oriented scenario method, which uses a GSS to mediate group work in the process and to enable efficient scenario creation. The method is introduced by Kivijärvi et al. [19] and later labeled the IDEAS method [27]. The mechanical details of the method have been described and discussed in detail in [19] and [25].

The often cited benefits of using a GSS are reduction of individual domineering, efficient parallel working, democratic discussion and decision making through anonymity on-line and voting tools, e.g. [13], [24]. These features are important as the subject matter of the process may be sensitive or controversial to some of the participants.

To illustrate the scenario process in general, and how the particular method we have chosen works, we will next walk through the main tasks of the scenario process (Fig. 1). The phases I-IV are completed in a group session under electronic mediation, preceded by common preparations and after the session the collected data is transformed to the final scenarios. The phases from III-post-phase can be also supported by mapping tools beside GSS.

The first main task during the process is to identify the drivers of change, the most influential players, change processes and other factors, which constrain and drive the development of the present. The second is to identify events, these drivers will trigger during the time span of the scenarios. As a third task, the group will assign an impact measure on the events based on how much they think the event will affect the organization or entity from whose point of view the scenarios look upon the future, and a probability measure to tell how probable the realization of each event is. These measures are used to group the events to sets as the fourth task, which make the scenarios. The grouping is inspected and discussed in the session and consistency of the events is inspected. The events and drivers will act as a base for the final scenario stories that will be written outside the session.

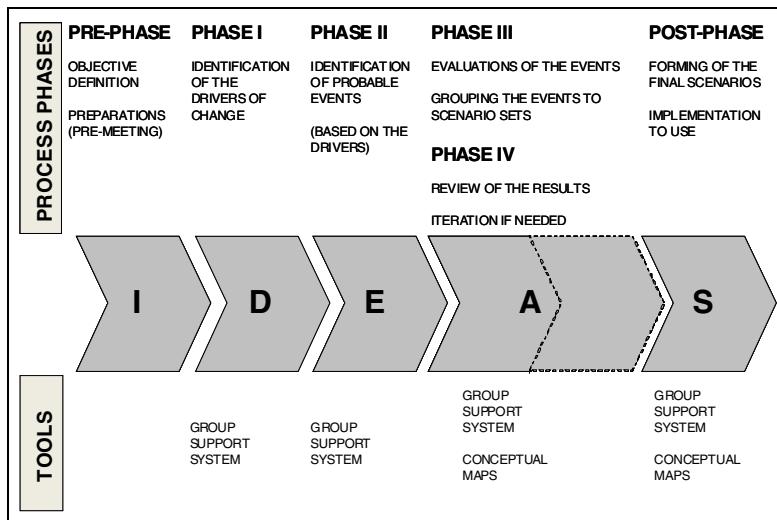


Fig. 1. The IDEAS scenario process and support tools [19], [27]

When we compare the process to the discussion about learning process and knowledge presented above, we can propose that the process follows the formula where the participants articulate their assumption when generating the drivers that change the world. The subsequent discussion will subject the assumptions to scrutiny and the group move toward new critically chosen set of assumptions when they vote for the most important drivers. Then they extrapolate assumptions when identifying the future events and when evaluating the events the participants effectively have to picture plausible actions and their effects. The final scenarios are presented outside the session.

### 3.2 Cases

The conceptual discussion above presented the premises for the argument that using a scenario process would form a community that encourages knowledge creation and sharing within an organization. To pave the way for the evaluation of our argument, we preset two concise case descriptions to illustrate the process. The first case focuses on strategic planning and positioning in a university [25]. The second case is taken from a project where the objective was to develop measures to identify and assess business opportunities at an intersection of industries [20], [26]. The cases both use the same process context although the communities are different.

The members of the semi-virtual community in the first case hold personal knowledge and experience in a number of areas such as research, teaching, and administration in different departments and in the administration of the whole university. The purpose was to discover new opportunities for the future position and operational environment of the university over the following ten years. The community was composed of individuals most of whom had met but who were not very familiar with each other. Thus, the most apparent link between most of the individuals was the presented problem of creating scenarios for the organization.

After the preparation, definition, and briefing of the problem, the actual work within the community started by brainstorming the key external uncertainties and drivers of change. The drivers form the backbone of the scenarios. This phase comprised an idea generation with a brainstorming tool, followed by a period for writing comments about the ideas and clarification of the proposed drivers. The discovered events they were grouped into initial scenarios by qualitative graphical clustering and discussed during the meeting. The GSS-workshop phase of the process ended in the evaluation of the events and graphical grouping, from which the data was moved to the remainder of the process.

The authors of the scenarios reflected on the cause and effect between drivers and events inside the scenario through systems thinking. Using systems analogy, the drivers of the scenarios form a system with feed-back relations, and the event are triggered by the interaction and feedback between the drivers. After mapping the drivers and the data cleanup, the events were organized into a concept map and tied together as logical chains with appropriate linking phrases; these described the connection and transition between the events. The names for the scenarios were picked after examining the general theme in the scenarios. In this case, in order to test the reactions and validate the logical structure of the maps, after the initial maps were drawn they were presented to some of the closer colleagues familiar with the sessions in the form of a focus group interview.

The final scenario stories were written around the logic of the concept maps. Other than some minor adjustment to the maps, the writing was a straightforward process of tying the events together as a logical story, from the present to a defined state in the future. The process might be characterized as iterative, a resonance between the drivers and the scenario maps conducted by the writer.

The purpose of the scenario process in the second case was to discover new opportunities at the intersection of a manufacturing and a complementary industry. For this case, the members of the semi-virtual community were selected from each industry, as well as from academics and general experts in the field. This time the people were from different organizations and had different backgrounds from research to management.

Generally, the process followed the same outline as described in the previous case. The process started with a briefing just the same as the fist case and the group engaged in fiding the drivers for the industry intersection. Based on the prioritized drivers the participants set out to identify new business ideas to the benefit of both industries, and thus were creating kinds of technology or business scenarios. After finding the events, the participants again evaluated the impact of the events but this time to the supplying industry as well as the target industry. Based on this evaluation the group evaluated the results and the session and the process moved out of the community and the scenario authors composed the final scenarios with the same basic tools as in the first case. In short, the process outline was similar and the community was able to produce plausible scenarios also in the second case, which essentially is a replication of the results from the first case.

Reportedly, the presented scenario method has served adequately in each context. However, the knowledge production properties have not been explicitly investigated in the reported cases beside some interviews administered to the participants in the first case. All in all, the participants of the sessions have generally reported the

approach as a viable tool for important decisions, even with its flaws. Additional finding has been that in addition to the concrete scenarios, some interviewees also saw the process as a kind of learning experience, promoting open-minded consideration of different options and ideas, and as a possibility to create consensus on large issues and goals in a large heterogeneous organization. The answer to the question of whether knowledge has been created is not as straightforward as satisfaction to the process. The interviewees were not confident to make strong statements to either direction, but they did indicate that thoughts and ideas were exchanged, which supports the proposition that knowledge was shared if not created. In any case, the results still point to the fact that the participants in the sessions were forming a community, exchanged, and diffused knowledge through the system, which in effect supports the argument in the paper. If we accept that conceptually scenarios are an embodiment of organizational knowledge, then a process which produces scenarios successfully indeed does create knowledge.

To support the conclusion further, some secondary evidence is available from Kokkonen et al. [20] as they report that compared to an established scenario method called Field Anomaly Relaxation, based on morphological analysis, the IDEAS scenarios are strongly attached to the views of the participants. Together with the fact that the reported satisfaction to the results and general buy-in to the scenarios is high, we can at least suggest that the scenarios done with the IDEAS method do have properties of organizational knowledge, as they are well accepted worldviews, or views to the world of tomorrow.

The main finding concerning the scenario process as a community for knowledge sharing and creation is that the people were able to work together as community and the process supported constructive discussion and engagement in both cases. Furthermore, both of the cases resulted in plausible and consistent scenarios. Based on these two cases we would like to conclude that the properties of the IDEAS method as a community can facilitate knowledge creation. However, we must leave a reservation that these conclusions are based on theoretical reasoning and two cases, and thus our results serve to highlight an interesting direction for further research in scenarios as both as a product and enabler of knowledge creation in organizations, rather than confirm such proposition.

The results may also apply to other scenario methods, as long as there is a group of people who actively participate in creating the scenarios, so that the conditions for community and knowledge can be satisfied. IDEAS is in that sense a representative example, because the main substance in the scenarios is essentially a product of group discussion, where the group expresses their views, discusses and reiterates the scenario material towards a consensus where they can agree upon the drivers and sets of events.

## 4 Discussion and Conclusions

We started the paper by arguing that scenarios are a piece of organizational knowledge and can be linked to knowledge creation in different levels. The main premise was that knowledge is capability to make decisions. A further premise is that the shared context can be provided in a community of practice, or in the absence of a community of practice, in a semi-virtual facilitated community. We also presented a

method to create scenarios and briefly examined two case studies which offer some support to our argument. Generally, the IDEAS method fulfills the conceptual requirements and the empirical experiences with the method suggest that the process is able to promote knowledge creation and sharing. Table 1 below condenses the conceptual discussion from section 2 to the epistemological criteria against which the scenario process as a context for knowledge creation and sharing can be evaluated, and illustrates the properties of the scenario process, particularly the IDEAS method, and how it fits to the criteria.

Examination of the findings suggested that the cases supported the theoretical propositions about supporting the semi-virtual community. In the light of the results, it seems that the concept of utilizing the supported scenario process to create actionable knowledge is feasible. Nevertheless, we would like to be cautious about drawing definite conclusions, but instead we would like to encourage further research in to knowledge creation in the scenario process and scenarios as a product of knowledge creation.

In the academic arena, the paper has contributed to the discussion about communities of practice and tested the use of communities for promoting knowledge creation. As for practical implications, the results suggest that the scenario process can facilitate integration and embodiment of organizational knowledge otherwise left tacit. This has in a sense rather direct implications for management, as engaging in the scenario process can have multiple benefits as discussed above.

The subject of scenarios as an embodiment of organizational knowledge can be studied further in a variety of directions. First of all we can study different kinds of scenario processes and different scenarios (see e.g. [3], [27] for discussion) beside the IDEAS method to learn which types support the process of knowledge creation and codify the knowledge to a usable form. The IDEAS method was used as an example to illustrate the scenario process and to gather some evidence to support the proposition, but it does not mean that IDEAS is the only scenario method that can support knowledge sharing. Another interesting question is that how much we can in fact know about the future, and how much scenarios are representations of current knowledge. Also the properties of scenarios as a way to rehearse for future actions would be an interesting subject for further study. To facilitate further study, Table 1 can be interpreted as a set of design propositions for the scenario process, which outline some of the meta-requirements [34] for supporting knowledge sharing and creation.

To conclude the paper, we propose that as far as knowledge is capability to make decisions, managers can raise their level of knowledge and capability to make decisions by undertaking the scenario process. Altogether, the case experiences suggest that the approach was at least partially able to engage the group in a semi-virtual community and to facilitate knowledge creation in the organizational context. The proposed scenario process seems to be a feasible way to integrate multidisciplinary groups to create knowledge in the form of the scenarios, which can be used to promote knowing future opportunities and decision options. The properties of scenarios promote and even require open minded consideration of the plausible beside the known and probable, which raises situation awareness and improves ability to act. With these conclusions, we would like to encourage further study into scenarios as a product and enabler of organizational knowledge creation.

**Table 1.** Evaluation and design propositions for the scenario process for knowledge creation and sharing

<b>Theoretical concept</b>	<b>Proposed properties that facilitate knowledge creation</b>
<i>Personal knowledge</i>	<i>The support system has to</i>
1. Object	Support in making categories and distinctions and organizing primary knowledge elements from the huge mass of knowledge and information overflow.
2. Path	Support creation of procedural knowledge by related steps.
3. Network	Help to create new relations between the knowledge elements and to relate participants over organization.
4. Tacit	Stimulate sharing and usage of tacit knowledge by providing a shared context for social processes; accepts personal experience.
5. Explicit	Support codification and sharing/diffusing of explicit knowledge assets.
6. Knowing	Integrates subjective, social, and physical dimensions of knowledge in the epistemic process of knowing. Support the interplay between the different types of knowledge and knowing.
<i>Organizational knowledge</i>	
1. Knowledge	Support creating organizational knowledge within the organization and with value chain partners.
2. Knowing	Support organizational decision making by applying organizational rules of actions.
<i>Context</i>	
1. Participation	Allow equal opportunity for participation.
2. Spontaneity	Diminish bureaucracy but allow to structure spontaneity. Keep the feeling of voluntarity.
3. Self-motivation	Support self-determination of goals and objectives. Allow the possibility to choose the time of participation. Explicate clear causality between personal efforts, group outcomes and personal outcomes.
4. Freedom from organizational constraints	Manage participants from different organizational units at various organizational levels.
5. Networking	Allow traditional face to face communication to promote mutual assurance between participants. Allows freedom of expression, verbal and non-verbal communication. Maintain social networking among participants.
<i>Scenario</i>	
1. Subsidiary awareness	Engage subsidiary and focal awareness of the past and future.
2. Focal awareness	
3. Foreseeing	Support the continuous process to integrate past, present and future.
4. Driver	Enable electronic discussion voting tools to identify of important drivers.
5. Event	Enable discussion and voting tools.
6. Chains of events	Provide maps and other representations to organize the knowledge of future drivers and events to scenarios.
7. Phases of the process	Accumulate information about the future and converge toward shared knowledge toward the end of the process.

## References

1. Amin, A., Roberts, J.: Knowing in action, beyond communities of practice. *Research Policy* 37, 353–369 (2008)
2. Bergman, J.-P.: Supporting Knowledge Creation and Sharing in the Early Phases of the Strategic Innovation Process. *Acta Universitatis Lappeenrantaensis* 212. Lappeenranta University of Technology, Lappeenranta (2005)
3. Bishop, P., Hines, A., Collins, T.: The current state of scenario development: an overview of techniques. *Foresight* 9(1), 5–25 (2007)
4. Blanning, R.W., Reinig, B.A.: A Framework for Conducting Political Event Analysis Using Group Support Systems. *Decision Support Systems* 38, 511–527 (2005)
5. Brown, J.S., Duguid, P.: Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation. In: Cohen, M.D., Sproull, L.S. (eds.) *Organizational Learning*, pp. 58–82. Sage Publications, Thousand Oaks (1996)
6. Brown, J.S., Duguid, P.: Knowledge and organization: A social-practice perspective. *Organization Science* 12(2), 198–213 (2001)
7. Brown, J.S., Duguid, P.: Structure and Spontaneity: Knowledge and Organization. In: Nonaka, I., Teece, D. (eds.) *Managing Industrial Knowledge*, pp. 44–67. Sage Publications, Thousand Oaks (2001)
8. Carlile, P., Rebentisch, E.S.: Into the Black Box: The Knowledge Transformation Cycle. *Management Science* 49(9), 1180–1195 (2003)
9. Chermack, T.J.: Improving Decision-Making with Scenario Planning. *Futures* 36, 295–309 (2004)
10. Chermack, T.J., van der Merwe, L.: The role of constructivist learning in scenario planning. *Futures* 35, 445–460 (2003)
11. Cook, S.D.N., Brown, J.S.: Bridging Epistemologies: The Generative Dance Between Organizational Knowledge and Organizational Knowing. *Organization Science* 10(4), 381–400 (1999)
12. Emery, J.C.: *Organizational Planning and Control Systems, Theory and Technology*. Macmillan Publishing Co. Inc., New York (1969)
13. Fjermestad, J., Hiltz, S.R.: Group Support Systems: A Descriptive Evaluation of Case and Field Studies. *Journal of Management Information Systems* 17(3), 115–159 (2001)
14. Gammelgaard, J., Ritter, T.: Virtual Communities of Practice: A Mechanism for Efficient Knowledge Retrieval in MNCs. *International Journal of Knowledge Management* 4(2), 46–51 (2008)
15. Jennex, M.E., Olfman, L.: A Model of Knowledge Management Success. *International Journal of Knowledge Management* 2(3), 51–68 (2006)
16. Kahn, H., Wiener, A.J.: *The Year 2000: A Framework for Speculation on the Next Thirty-Three Years*. Collier-Macmillan Ltd., London (1967)
17. Kimble, C., Hildred, P., Wright, P.: Communities of Practice: Going Virtual. In: Malhotra, Y. (ed.) *Knowledge Management and Business Model Innovation*, pp. 220–234. Idea Group Inc., Hershey (2001)
18. Kivijärvi, H.: Aligning Knowledge and Business Strategies within an Artificial Ba. In: Abou-Zeid, E.-S. (ed.) *Knowledge Management and Business Strategies: Theoretical Frameworks and Empirical Research*. Idea Group Inc., Hershey (2008)
19. Kivijärvi, H., Piirainen, K., Tuominen, M., Elfvengren, K., Kortelainen, S.: A Support System for the Strategic Scenario Process. In: Adam, F., Humphreys, P. (eds.) *Encyclopedia of Decision Making and Decision Support Technologies*, pp. 822–836. Idea Group Inc., Hershey (2008)

20. Kokkonen, K., Piirainen, K., Kässi, T.: E-business opportunities in the Finnish forest sector – a multi-method scenario study. In: The Proceedings of XVIII International Conference of International Society for Professional Innovation Management, Tours, France (2008)
21. Lave, J., Wenger, E.: Situated learning: Legitimate peripheral participation. Cambridge University Press, New York (1991)
22. Lesser, E., Everest, K.: Using Communities of Practice to Manage Intellectual Capital. *Ivey Business Journal* 65(4), 37–41 (2001)
23. Leonard, D., Sensiper, S.: The Role of Tacit Knowledge in Group Innovation. *California Management Review* 40(3), 112–132 (1998)
24. Nunamaker Jr., J.F., Briggs, R.O., Mittleman, D.D., Vogel, D.R., Balthazard, P.A.: Lessons from a Dozen Years of Group Support Systems Research: A Discussion of Lab and Field Findings. *Journal of Management Information Systems* 13(3), 163–207 (1997)
25. Piirainen, K., Tuominen, M., Elfvingren, K., Kortelainen, S., Niemistö, V.-P.: Developing Support for Scenario Process: A Scenario Study on Lappeenranta University of Technology from 2006 to 2016, Research report 182. Lappeenranta University of Technology, Lappeenranta (2007), <http://urn.fi/URN:ISBN:978-952-214-369-3>
26. Piirainen, K., Kortelainen, S., Elfvingren, K., Tuominen, M.: A scenario approach for assessing new business concepts. *Management Research Review* 33(6), 635–655 (2010)
27. Piirainen, K., Lindqvist, A.: Enhancing business and technology foresight with electronically mediated scenario process. *Foresight* 12(2), 16–37 (2010)
28. Polanyi, M.: Personal Knowledge. University of Chicago Press, Chicago (1962)
29. Polanyi, M.: The Tacit Dimension. Doubleday & Company Inc., Reprinted Peter Smith, Gloucester (1966)
30. Schwartz, P.: The Art of the Long View: Planning for the Future in an Uncertain World. Doubleday Dell Publishing Inc., New York (1996)
31. Simon, H.A.: The New Science of Management Decisions. Harper Brothers, New York (1960)
32. Swan, J., Scarborough, H., Robertson, M.: The Construction of ‘Communities of Practice’ in the Management of Innovation. *Management Learning* 33(4), 477–496 (2002)
33. Tsoukas, H., Vladimirov, E.: What is Organizational Knowledge? *Journal of Management Studies* 38(7), 973–993 (2001)
34. Walls, J.G., Widmeyer, G.R., El Sawy, O.A.: Building an Information Systems Design Theory for Vigilant EIS. *Information Systems Research* 3(1), 36–59 (1992)
35. Wright, G., van der Heijden, K., Burt, G., Bradfield, R., Cairns, G.: Scenario Planning Interventions in Organizations: An analysis of the causes of success and failure. *Futures* 40, 218–236 (2008)
36. Zeleny, M.: Human Systems Management: Integrating Knowledge, Management and Systems. World Scientific Publishing, Singapore (2005)

# Neural Networks Aided Automatic Keywords Selection

Błażej Zyglarski<sup>1</sup> and Piotr Bała<sup>1,2</sup>

<sup>1</sup> Faculty of Mathematics and Computer Science, Nicolaus Copernicus University  
Chopina 12/18, 87-100 Toruń, Poland

<sup>2</sup> ICM, Warsaw University, ul.Pawinskiego 5a, Warsaw, Poland  
[bzstyler@mat.umk.pl](mailto:bzstyler@mat.umk.pl), [bala@mat.umk.pl](mailto:bala@mat.umk.pl)  
<http://www.mat.umk.pl/~bzstyler>

**Abstract.** Document analysis is always connected with proper selection of keywords. Many different techniques were developed. This article presents our approach to keywords generation. This approach uses Kohonen Self Organizing Maps extended with reinforcement based on large library of documents, which are divided into categories (described in another articles). This article shows the differences between this method and statistical method and neural network without reinforcement. In the ending we discuss problem of algorithm performance, which can be improved by using Genetic Algorithms, median approximation and shape-changing neural networks.

## 1 Introduction

This article contains complete description and comparison of two approaches of keywords choosing: the statistical one and the neural network based one. Second one consists of two types: the simple neural network and the neural network with the reinforcement. In both cases we are using as the analysis example the same input string built over a two letters alphabet, which is "aab abb abaa abb bbba abb bbaa baa bbaa bbbb bbba bbbb". The last part of this document shows effectiveness of the reinforced learning and other algorithms. All presented approaches was tested with example set of about 200 documents.

## 2 Statistical Approach

The standard approach to selecting keywords is to count their appearance in the input text. As it was described in Narimura, most of frequent words are irrelevant. The next step is deleting trash words with use of specific word lists.

In order to speed up the trivial statistical algorithm for keywords generation, we've developed the algorithm which constructs the tree for text extracted from a document. The tree is built letter by letter and each letter is read exactly once, which guarantees that the presented algorithm works in a linear time.

### 2.1 Algorithm

Denote the plain text extracted from an  $i$ -th document as  $T(i)$ . Let  $\hat{T}(i)$  be a lowercase text devoid of punctuation. In order to select appropriate keywords, we need to build a tree (denoted as  $\tau(\hat{T}(i))$ ).

1. Let  $r$  be a root of  $\tau(\widehat{T}(i))$ . Let  $p$  be a pointer, pointing at  $r$

$$p = r \quad (1)$$

2. Read a letter  $a$  from the source text.

$$a = \text{ReadLetter}(\widehat{T}(i)) \quad (2)$$

3. If there exists an edge labeled by  $a$ , leaving a vertex pointed by  $p$ , and coming into some vertex  $s$  then

$$p = s \quad (3)$$

$$\text{count}(p) = \text{count}(p) + 1 \quad (4)$$

else, create a new vertex  $s$  and a new edge labeled by  $a$  leaving vertex  $p$  and coming into vertex  $s$

$$p = s \quad (5)$$

$$\text{counter}(p) = 1 \quad (6)$$

4. If  $a$  was a white space character then go back with the pointer to the root

$$p = r \quad (7)$$

5. If  $\widehat{T}(i)$  is not empty, then go to step 2.

After these steps every leaf  $l$  of  $\tau(\widehat{T}(i))$  represents some word found in the document and  $\text{count}(l)$  denotes appearance frequency of this word. In most cases the most frequent words in every text are irrelevant. We need to subtract them from the result. This goal is achieved by creating a tree  $\rho$  which contains trash words.

1. For every document  $i$

$$\rho = \rho + \tau(\widehat{T}(i)) \quad (8)$$

Every time a new document is analyzed by the system,  $\rho$  is extended. It means, that  $\rho$  contains most frequent words all over files which implies that these words are irrelevant (according to different subjects of documents). The system is also learning new unimportant patterns. With increase of analyzed documents, accuracy of choosing trash words improves.

Let  $\Theta(\rho)$  be a set of words represented by  $\Theta(\rho)$ . We should denote as a trash word a word, which frequency is higher than median  $m$  of frequencies of words which belongs to  $\Theta(\rho)$ .

1. For every leaf  $l \in \tau(\widehat{T}(i))$  If there exist  $z \in \rho$ , such as  $z$  and  $l$  represents the same word and  $\text{count}(z) > \text{count}(m)$  then

$$\text{count}(l) = 0 \quad (9)$$

## 2.2 Limitations

Main limitation of this type of generating keywords is loosing the context of keywords occurrences. It means that words which are most frequent (which implies that they are appearing together on the result list) could be actually not related. The improvement for this issue is to pay the attention at the context of keywords. It could be achieved by using neural networks for grouping words into locally closest sets. In our approach we can select words which are less frequent, but their placement indicates, that they are important. Finally we can give them a better position at the result list.

## 3 Kohonen's Neural Networks Approach

During implementation of the document management system [6] we've discovered previously mentioned limitations in using simple statistical methods for keywords generation. Main goal of further contemplations was to pay attention of the word context. In other words we wanted to select most frequent words, which occur in common neighborhood. Presented algorithm selects the most frequent words, which are close enough. It is achieved with a words categorization [3].

Precise results are shown in the Table 1. Each keyword is presented as a pair (a keyword, a count). Keywords are divided into categories with use of Kohonen neural networks. Each category has assigned rank, which is related to number of gathered keywords and their frequency).

**Table 1.** Example keywords selection

Category	Rank	Keyword	Count
1	0,94	keywords	8
		discovery	3
		frequent	3
		simple	2
		text	1
		automatic	1
2	0,58	neural	2
		networks	2
		discovered	1
		neighborhood	1
3	0,50	fulfill	1
4	0,41	statistical	2
		words	2
		contemplations	1
		promotion	1
		disadvantages	1
5	0,36	retrieve	1
		reliable	1

Results were filtered using words from  $\rho$  defined in first part of this article.

### 3.1 Algorithm

Algorithm of neural networks based keyword discovery consists of 3 parts. At the beginning we have to compute distances between all words. Then we have to discover categories and assign proper words to them. At the end we need to compute the rank od each category.

**Counting Distances.** Lets  $\widehat{T}^f = \widehat{T}(f)$  be a text extracted from a document  $f$  and  $\widehat{T}^f(i)$  be a word placed at the position  $i$  in this text. Lets denote the distance between words  $A$  and  $B$  within the text  $\widehat{T}^f$  as  $\delta^f(A, B)$ .

$$\delta^f(A, B) = \quad (10)$$

$$= \min_{i,j \in \{1, \dots, n\}} \{\|i - j\|; A = \widehat{T}^f(i) \wedge B = \widehat{T}^f(j)\} \quad (11)$$

By the position  $i$  we need to understand a number of white characters read so far during reading text. Every sentence delimiter is treated like a certain amount  $W$  of white characters in order to avoid combining words from separate sequences (we empirically chose  $W = 10$ ). Weve modified previously mentioned tree generation algorithm. Analogically we are constructing the tree, but every time we reached the leaf (which represents a word) we are updating the distances matrix  $M$  (presented on Fig. 1), which is  $n \times n$  upper-triangle matrix and  $n \in N$  means number of distinct words read so far.

Generated tree (presented on Fig. 2) is used for checking previous appearance of actually read word, assigning the word identifier (denoted as  $\xi^f(j) = \xi(T^f(j))$ ,  $\xi^f(j) \in N$ ) and to decide whether update existing matrix elements or increase matrix's dimension by adding new row and new column. Fig. 1 marks with dashed borders fields updateable in the first case. For updating existing matrix elements and counting new ones we also use the array with positions of last occurrences of all read words. Lets denote this array as  $\lambda(\xi^f(j))$ . We need to consider two cases:

1. Lets assume that  $j - 1$  words was read from input text and a word with position  $j$  is read for the first time.

$$\forall_{i < j} \widehat{T}^f(j) \neq \widehat{T}^f(i) \quad (12)$$

$$\lambda(\xi^f(j)) = j \quad (13)$$

$$\forall_{k \in \{1, \dots, \xi^f(j)\}} M[k, \xi^f(j)] = |j - \lambda(k)| \quad (14)$$

This case is shown on Fig. 3 and Fig. 4, which contains a visualization of the state of am algorithm after reading three first words from example string "aab abb abaa — abb bbba abb bbaa baa bbaa bbbb bbba bbbb". At the Fig. 4 there is also presented the table of last occurrences.

2. Lets assume that  $j - 1$  words was read from input text and a word with position  $j$  was read earier and already has given a worde identifier.

$$\exists_{i < j} \widehat{T}^f(j) = \widehat{T}^f(i) \quad (15)$$

We have to update the last occurrences table by

$$\lambda(\xi^f(j)) = j \quad (16)$$

and update appropriate row and column in existing matrix

$$\forall_{k \in \{1, \dots, \xi^f(j)-1\}} M[k, \xi^f(j)] = \Delta(k, \xi^f(j)) \quad (17)$$

$$\forall_{k \in \{\xi^f(j)+1, \dots, \hat{\xi}^f\}} M[\xi^f(j), k] = \Delta(\xi^f(j), k) \quad (18)$$

where

$$\hat{\xi}^f = \max_{i \in \{1, \dots, j\}} \{\xi^f(i)\} \quad (19)$$

$$\Delta(k, l) = \min(|\lambda(k) - \lambda(l)|, M[k, l]) \quad (20)$$

This case is shown on Fig. 5 and Fig. 6, which contains visualization of the state of the algorithm after reading four first words from an example string "aab abb abaa abb — bbba abb bbaa baa bbaa bbbb bbba bbbb". In this case a word "abb" was read twice, so we need to update  $\lambda$  array and the actual distance matrix  $M$ .

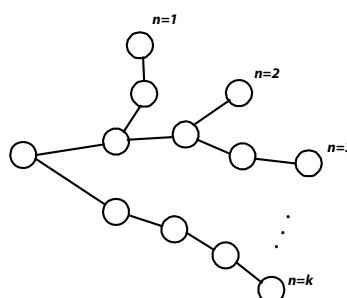
Bold-faced fields indicates values, that could be evaluated each time using  $\lambda$  array and dont have to be memorized. According to that observation we can omit the distance table, using instead of it only specific lists (Fig. 9), containing these elements, which cannot be evaluated with  $\lambda$  array. Final results of our example (after reading all words from string "aab abb abaa abb bbba abb bbaa baa bbaa bbbb bbba bbbb —") are shown on Fig. 7 and Fig. 8.

Presented algorithm guarantees that its result matrix is a matrix of shortest distances between words.

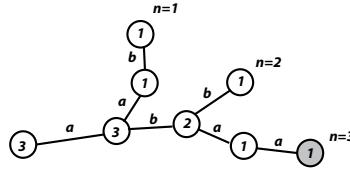
$$\delta^f(A, B) = M[\xi(A), \xi(B)] \quad (21)$$

$\xi^f$	1	2	...	$k$	...	$n$
$I$	0			*		
2		0		*		
...			...	*		
$k$				0*	*	*
...					...	
$n$						0

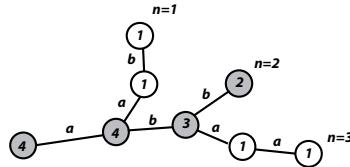
**Fig. 1.** Distances matrix



**Fig. 2.** Word tree

**Fig. 3.** The word tree generated after reading first three words

$\lambda(\xi^f)$	1	2	3
$\xi^f$	1	2	3
aab	1	0	1
abb	2	0	1
abaa	3	0	

**Fig. 4.** The distance matrix generated after reading first three words**Fig. 5.** The word tree generated after reading first four words

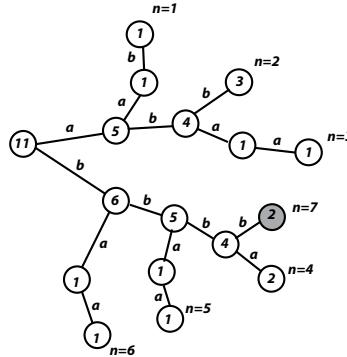
$\lambda(\xi^f)$	1	4	3
$\xi^f$	1	2	3
aab	1	0	1
abb	2	0	1
abaa	3	0	

**Fig. 6.** The distance matrix generated after reading first four words

**Generating Categories.** The knowledge of distances between words in document is used to categorize them with use of the self-organizing Kohonen Neural Network (Fig 10) [4].

This procedure takes 5 steps:

1. Create rectangular  $m \times m$  network, where  $m = \lfloor \sqrt[4]{\hat{\xi}^f} \rfloor$ . Presented algorithm can distinguish maximally  $m^2$  categories. Every node (denoted as  $\omega_{x,y}$ ) is connected with four neighbors and contains a prototype word (denoted as  $\hat{T}_\omega^f(x, y)$ ) and a set (denoted as  $\beta_\omega^f(x, y)$ ) of locally close words.
2. For each node choose random prototype of the category  $p \in \{1, 2, \dots, \hat{\xi}^f\}$ .
3. For each word  $k \in \{1, 2, \dots, \hat{\xi}^f\}$  choose closest prototype  $\hat{T}_\omega^f(x, y)$  in network and add it to list  $\beta_\omega^f(x, y)$ .



**Fig. 7.** The word tree generated after reading all words

	$\lambda(\xi^f)$	1 6 3 11 9 8 12
	$\xi^f$	1 2 3 4 5 6 7
aab	1	0 1 <b>2</b> 4 6 7 9
abb	2	0 1 1 1 2 4
abaa	3	0 2 4 5 7
bbba	4	0 <b>2</b> 3 <b>1</b>
bbaa	5	0 <b>1</b> 1
baa	6	0 2
bbbb	7	0

**Fig. 8.** The distance matrix generated after reading all words

$\lambda(\xi^f)$	$\lambda(1)$	$\lambda(2)$	...	$\lambda(\hat{\xi}^f)$
$\xi^f$	$I$	2	...	$\hat{\xi}^f$
	$\delta^f(1, u_{1,1})$	$\delta^f(1, u_{2,1})$	...	$\delta^f(1, u_{\hat{\xi}^f,1})$
	$\delta^f(1, u_{1,2})$	$\delta^f(1, u_{2,2})$	...	$\delta^f(1, u_{\hat{\xi}^f,2})$
	...	...	...	...
	$\delta^f(1, u_{1,k_1})$	$\delta^f(1, u_{2,k_2})$	...	$\delta^f(1, u_{\hat{\xi}^f, k_{\hat{\xi}^f}})$

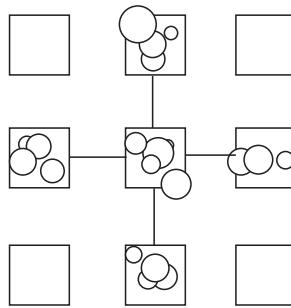
**Fig. 9.** The scheme of the distance table

- For each network node  $\omega_{x,y}$  compute a generalized median for words from  $\beta_\omega^f(x, y)$  and neighbors lists (denoted as  $\beta$ ). A generalized median is defined as an element  $A$  which minimizes a function:

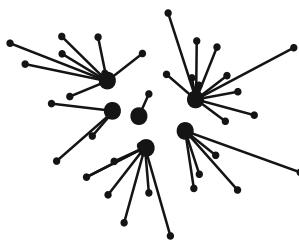
$$\sum_{B \in \beta} \delta^{f2}(A, B) \quad (22)$$

Set  $\widehat{T}_\omega^f(x, y) = A$

- Repeat step 4 until the network is stable. Stability of the network is achieved, when in two following iterations all word lists are unchanged (without paying attention to internal lists structure and their position in nodes).



**Fig. 10.** The scheme of the neural network for words categorization with selected element and its neighbors



**Fig. 11.** The two dimensional example of Kohonen network grouping locally closest words (small circles) into different categories

Such algorithm divides the set of all words found in document into separate subsets, which contains only locally close words. In other words it groups words into related sets (see Fig. 11).

**Selecting Keywords.** After execution of described procedure all words from the analyzed document are divided into categories. We need to choose most important categories and then select most frequent words among them. Each category contains list of words, which are locally close. As a category rank we could take

$$\rho_{x,y} = \frac{\sum_{w \in \beta_\omega^f(x,y)} \text{count}(w)}{|\beta_\omega^f(x,y)|} \quad (23)$$

Now ordering categories descending by  $\rho_{x,y}$  we can select from each a number of most frequent words. This method allows selecting words, which appearance isn't most frequent in the complete scope of a document, but is frequent enough in some sub-scope. The sample results are shown in table 1. The table presents first eighteen results presented method.

## 4 Neural Network Reinforcement

At this stage of work we've decided to extend an idea of presented algorithm with use of a reinforcement for improving it's accuracy. The reinforcement is performed with

**Table 2.** The comparison of results for the example text

	Statistical	Kohonen	Reinforced
<i>words</i>	40	<i>words</i>	40
<i>keywords</i>	25	<i>keywords</i>	25
<i>text</i>	22	<i>neural</i>	13
<i>neural</i>	13	<i>distances</i>	3
<i>abb</i>	13	<i>text</i>	22
$\widehat{T}^f$	11	<i>simple</i>	5
<i>matrix</i>	11	<i>elements</i>	2
<i>tree</i>	9	<i>reading</i>	8
<i>frequent</i>	9	<i>extracted</i>	3
$\widehat{T}$	9	$\widehat{T}^f$	11
<i>bbbb</i>	8	<i>matrix</i>	11
<i>networks</i>	8	<i>distance</i>	7
<i>extracted</i>	3	<i>discovery</i>	5
		<i>keyword</i>	25
		<i>distance</i>	7
		<i>neural</i>	13
		<i>statistical</i>	10
		<i>context</i>	4
		<i>matrix</i>	11
		<i>denoted</i>	4
		<i>vertex</i>	3
		<i>web</i>	2
		<i>relevant</i>	2
		<i>string</i>	3
		<i>extracted</i>	3
		<i>networks</i>	8

use of presented in [6] system, in which we have created huge repository of documents. Articles used for testing algorithms were also put into this repository. It means that all of them were compared using statistics of words, statistics of n-grams [1] (in this particular case 3-grams) and Kolmogorov Complexity [2]. Three kinds of distances between documents are computed.

#### 4.1 Reinforcement

Main idea of the reinforcement [5] is to modify a behavior of the neural network depending of a weight of a keywords candidate. At the beginning we need to initiate the weight attribute (with values from interval  $[0,1]$ ) of every word from a document. Each word, which can be found in previously mentioned trash word set has weight equal to 0, rest of them have weights equal to 1. The neural network algorithm is modified in such way, that words with smaller weight are pushed away from words with greater weight. It means that they are pushed aside the main categories. Of course they will have also small influence on category rank. Moreover we need to add parent iteration, which will modify weights of words and repeat neural network steps until proper words will be selected. After performance of word categorization a set of proposed keywords is generated. At this stage we need to check every keyword for it's accuracy. This is performed by checking a number (in our tests it was 10) of articles (containing tested keyword) randomly selected from repository and comparing normalized distances between them and the analyzed document. If these documents are relatively close (in the terms of counted distances) to initial one, a keyword is prized with increasing it's weight. If distances are relatively far, weight is decreased. In other words, if an selected keyword is good, a network is rewarded. With this improvement, algorithm continues with steps of neural network learning.

#### 4.2 The Results Propriety

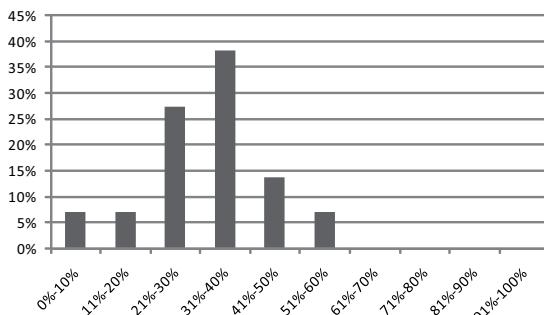
Methodology of creating repository, which is described in [6] guarantees, that collected documents has various subjects. They are gathered with using of most frequent words

appeared in each document. Additional variety is a result of the collection which initiates repository - containing articles from various areas of interests. It means that there is a big chance, for articles containing tested keyword to be really connected with the same subject. If a keyword candidate isn't really a keyword, these documents will probably differ from tested one and network will not be reinforced.

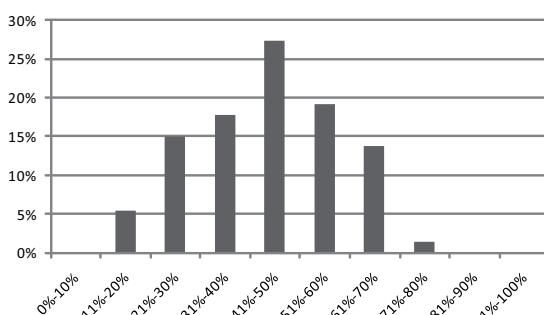
## 5 The Comparison of Results

Presented method gives better results than the simple statistical method. In table 2 we show keywords found over this article, chosen with using all three methods (with italic font there are marked actual (subjectively selected by authors) keywords). It's clear that Kohonen Networks related methods gives better results than statistical method and also reinforcement has very good influence on final results.

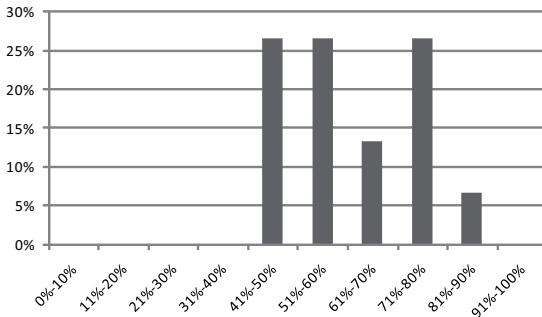
In our tests we've used about 200 various articles. In most cases results given by our approach was more accurate than other approaches. The accuracy was checked manually and is subjective. Finally, according to executed tests, statistical methods gave very poor results (see Fig. 12). In the best case list of proposed keywords achieved



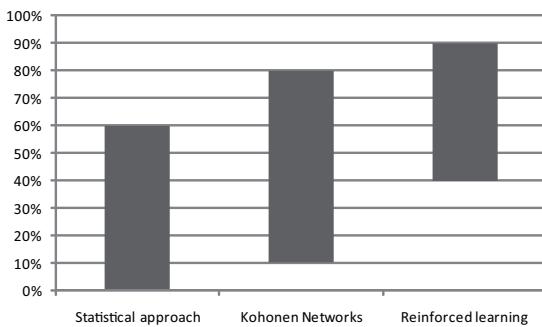
**Fig. 12.** Effects of statistical method. X axis shows effectiveness, Y axis shows number of documents with processed with this effectiveness.



**Fig. 13.** Effects of neural network method. X axis shows effectiveness, Y axis shows number of documents with processed with this effectiveness.



**Fig. 14.** Effects of neural network method with reinforcement. X axis shows effectiveness, Y axis shows number of documents with processed with this effectiveness.



**Fig. 15.** Comparison of presented methods

65% accuracy. In the worst case it was about 5%. At the Fig. 12 there are presented accuracies of results from tested articles (for example: in 66% of articles accuracy of keywords was at level between 20% and 40%). Better results archived with Kohonen Networks without the reinforcement are presented at the Fig. 14). In the best case, list of proposed keywords achieved almost 80% accuracy. 10%-40% accuracy was in this case very less often. The best result were generated with using Reinforced Kohonen Networks, where best results reached level of 90% accuracy and results between 10%-40% were completely eliminated. Comparison of effectiveness of all presented methods is shown on Fig. 15.

## 6 Document Management System Implementation

The Web Services based Scientific Article Manager [6] has been designed to provide the knowledge management for the users using journal articles and internet documents as main sources of information. The journal articles are usually retrieved in PDF format, however the content can be processed and analyzed automatically (excluding relatively small number of documents which are stored as direct scans). During selection of documents connected to examined article, there are used three different methods of comparison, which are finally combined. It guarantees accuracy of obtained results, which pay attention to content and structure of documents:

1. Statistical based, performed by checking appearance of all words in two documents. Each word is treated as a dimension and a taxicab metric is used for counting distances between documents. In this case all punctuation and white characters are excluded.
2. N-grams based, performed by checking appearance of all n-grams in two documents. Each n-gram is treated as a dimension, analogically to previous method.
3. Kolmogorov distance based. Kolmogorov complexity of text  $X$  (denoted as  $K(X)$ ) is length of the shortest compressed binary file  $X^*$ , from which original text can be reconstructed. Formally Kolmogorov Complexity  $K(x)$  is defined as length of the smallest program running on Turing Machine, which returns word  $x$ . In our case Turing Machine is substituted by compression program and compressed file responds to the Turing program.

Presented system is implemented with Java language and provides data with use of the webservices technology. We've also implemented three kinds of user interfaces:

1. Servlet based;
2. Portlet based (used with Liferay portal);
3. .Net Windows client (used with Windows explorer, in future intended to work seamlessly within Windows explorer)

The users are about to share, and access each other documents, with appropriate permission subsystem.

## 7 Performance

Kohonen categorization can be fastened and improved by substitution of median computation with Genetic Algorithms and making neural network shape-changing. In order step of median computation and network reorganization (4,5) should be replaced by:

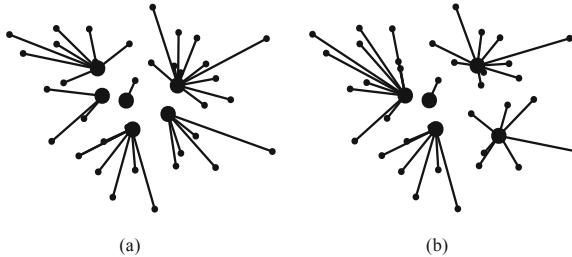
1. Nodes Self-organization (Genetic Algorithms),
2. Empty nodes deletion,
3. Median Approximation.

For further research type of the network should be changed. As a neighbor nodes, nodes with closest prototypes should be used.

Nodes Self-organization. Genetic algorithms are based on evolution theory, which assumes, that organisms living in some environment reproduce and give life to successors, who fit better into this environment. Worse organisms die faster. Best organisms live longer. Each organism is treated as a gene and can be used in major operations like mutation, crossover and selection. While performing this step, elements from each node list should be moved to the closest better node (if such exists) - it can be understood as crossover.

Empty nodes deletion. If some nodes became empty (it means that there is no words in word lists), they should be deleted and its prototypes should be reinserted into other nodes. It reduces number of computations because of minimizing of network size.

**Median Approximation.** Assuming that all nodes are organized, we can approximate median with computing it only for nodes prototypes. It also reduces significantly number of computations. All results of modified algorithm correspond (see Fig. 16) to results of previously presented algorithm. Using it gives faster as good keywords selection as previously.



**Fig. 16.** Comparison of one step results of standard (a) and modified (b) algorithm

## Notes and Comments

Presented approach gives in most cases very good results of the context aware keywords recognition. Effectiveness is related with size of the repository of available documents, which are used for reinforcement of the neural network. It means, they are more effective while working with large data collections. Moreover continuously working Kohonen network can improve the keywords recognition every time new documents are added to the repository. If neural network is stable, any change of keywords weights implies reorganisation of this network, which (according to size of performed changes - only weights in proposed keywords are changed) is in most cases quite fast. Most of the processing time is spent to check accuracy of proposed keywords. Fortunately, results of this check are used also for categorization of documents collected in our Document Management System.

## References

1. Cavnar, W.B., Trenkle, J.M.: N-gram-based text categorization. In: Proceedings of SDAIR 1994, 3rd Annual Symposium on Document Analysis and Information Retrieval, pp. 161–175 (1994)
2. Fortnow, L.: Kolmogorov complexity and computational complexity (2004)
3. Frank, E., Chui, C., Witten, I.H.: Text categorization using compression models. In: Proceedings of DCC 2000, IEEE Data Compression Conference, Snowbird, US, pp. 200–209 (2000)
4. Kohonen, T.: The self-organizing map. Neurocomputing 21(1-3), 1–6 (1998)
5. Sutton, R.S.: Generalization in reinforcement learning: Successful examples using sparse coarse coding. In: Advances in Neural Information Processing Systems 8, vol. 8, pp. 1038–1044 (1996)
6. Zyglarski, B., Baa, P., Schreiber, T.: Web services based scientific article manager. In: Information Systems Architecture and Technology, Web Information Systems: Models, Concepts and Challenges, pp. 205–215 (2008)

# Value Knowledge Management for Multi-party Conflicts: An Example of Process Structuring

Shahidul Hassan and John Rohrbaugh

Rockefeller College of Public Affairs, University at Albany, SUNY  
1400 Washington Avenue, Albany, NY 12202, U.S.A.  
[{sh751124,jwr26}@albany.edu](mailto:{sh751124,jwr26}@albany.edu)

**Abstract.** Value knowledge management (VKM) comprises the process structuring required to make individual and/or group values explicit in a manner so that such initially tacit knowledge appropriately informs decision making. This paper presents a case in which VKM is used for structuring an organizational preparation process for a new and substantial initiative. Fundamental group conflicts exist with respect to this initiative and, more immediately, with respect to the extent of preparation envisioned. The relative importance of two key values is at issue: increasing human capital and reducing project costs. The case illustrates a three-stage approach to VKM and demonstrates how the articulation of group judgment policies, the development of a shared resource allocation model, and the application of analytical mediation can make a substantial contribution to organizational problem solving or opportunity seeking.

**Keywords:** Knowledge management, Values, Judgment analysis, Resource allocation, Analytical mediation.

## 1 Introduction

In the field of knowledge management (KM), the distinction between tacit and explicit knowledge has remained an important touchstone [1]. While explicit knowledge has been articulated, codified, and communicated already in some symbolic form, tacit knowledge, though perhaps equivalent in its coherence and correspondence [2], remains as yet implicit and unexpressed. Tacit knowledge must be inferred by others over time as actions are observed. Both individuals and groups are viewed as possessing tacit knowledge; some have argued that organizations also can be considered to be repositories of tacit, as well as explicit, knowledge [3].

One of the most important domains of tacit knowledge pertains to values, that is, personal values, group values, and organizational values. According to Scott [4], a value is a standard which influences—in full or in part—commitment to preferred actions and goals (i.e., what ought to be accomplished or what ought to be achieved). When one value alone fully explains commitment to an action or goal (e.g., the standard for preserving all human life or for speaking only the truth), this value is absolute. In most situations, however, two or more relative or competing values differentially influence such commitments.

Surprisingly, value knowledge is not identified as a type (e.g., declarative, procedural, causal, conditional, relational, or pragmatic) in knowledge taxonomies [5]. Value knowledge management (VKM), a proposed domain for KM introduced in the present paper, is absolutely central to any explication of organizational problem solving or opportunity seeking. VKM comprises the process structuring required to make individual and/or group values explicit in a manner so that such initially tacit knowledge appropriately informs decision making and provides necessary retraceability and sufficient accountability. Without VKM, an organization is unable to maintain its intentional course because it lacks capacity either to articulate or to exercise its priorities.

Values cannot be articulated meaningfully in the abstract, of course, and any general statement of their relative importance is useless [6]. Therefore, the foundation of VKM is the assumption that the most informative expression of individual and group values always will be in reference to specific and well-understood situations. The management of value knowledge must originate in particular circumstances that can elicit statements of preference. Since values are the standards which influence commitment to preferred actions and goals, the clearest insight into their relative importance—if trade-offs are induced at all—emerges where they are “put to the test”.

The present paper presents a case in which VKM is used to structure an organizational preparation process for a new and substantial initiative. Fundamental group conflicts exist with respect to this initiative and, more immediately, with respect to the extent of preparation envisioned. The relative importance of two key values is at issue: increasing human capital and reducing project costs. In turn, these two values influence the level of individual and group commitment to five preferred organizational actions: process planning, process scope, process staffing, trainer skill, and suitability of facilities. In this case, VKM entails a sequence of three stages: the articulation of group judgment policies, the development of a shared resource allocation model, and the application of analytical mediation.

## 2 Group Judgment Policies

One of the most well-tested and applied methods for measuring individual and group commitment to preferred actions and goals is through the use of judgment analysis [7, 8]. Sometimes identified as “policy capturing,” judgment analysis typically involves the presentation of a series of realistic cases, scenarios, or vignettes that systematically differ on several well-specified dimensions. By regressing numerical judgments that are expressed in response to variations in these dimensions, an explicit model of the judgment process can be inferred that algebraically represents—and can predict—the assessments made in a judgment process.

In the present case, five dimensions of organizational action are contemplated: process planning, process scope, process staffing, trainer skill, and suitability of facilities. The judgment to be made is the extent to which these dimensions influence increases in human capital of relevance to the new and substantial initiative.<sup>1</sup> Three groups—teams from human resources management (HRM), budget and finance (B&F), and new project coordination (NPC)—with long-standing conflicts of value

---

<sup>1</sup> Reductions in project costs are considered in the next section.

within the organization independently meet in a brief session to articulate their respective judgment policies.

The initial series of 35 hypothetical scenarios presented to each group for consideration is illustrated by three cases shown in Figure 1; a full description of the method is beyond the scope of this paper [see 9]. The relative weights and function forms that the three groups produce for the five dimensions of organizational action are displayed in Figure 2. Note, for example, that HRM places the greatest relative weight on planning, which is least important to B&F. Both function forms for the dimension of facilities are positive for HRM and NPC; B&F, however, generates a negative function form. Even in this first stage of VKM, these three sets of relative weights and function forms make explicit the nature of the organizational conflict that exists between the three groups.

<b>Case 1</b>
Planning Level 3: one-day meeting on-site
Scope Level 2: 25 participants; 6 one-day sessions
Staffing Level 3: full-time manager
Trainer Level 4: regional contractor
Facilities Level 1: in-house space
<b>Case 2</b>
Planning Level 5: two-day meeting off-site
Scope Level 3: 15 participants; 4 two-day sessions
Staffing Level 2: half-time manager
Trainer Level 4: regional contractor
Facilities Level 3: conference center
<b>Case 3</b>
Planning Level 3: one-day meeting on-site
Scope Level 1: 15 participants; 6 one-day sessions
Staffing Level 1: half-time senior clerical
Trainer Level 2: in-house staff with consultant
Facilities Level 2: in-house space with catering

**Fig. 1.** Examples of three scenarios presented for group judgments

### 3 Shared Resource Allocation Model

A resource allocation model identifies the full set of activities, projects, or programs vying for support, as well as the multiple levels at which investments could be made in each. A full description of the method for constructing resource allocation models with groups is also beyond the scope of this paper [see, for example, 10, 11, 12, 13, 14]. The shared resource allocation model for the present case is presented in Figure 3. Five levels of resource investments are being considered for each organizational action; levels are listed from left to right across the rows in order of their increasing costs as the B&F team estimates.<sup>2</sup>

<sup>2</sup> In cases where teams disagree on cost projections, additional meetings to achieve consensus may be required. The use of “sensitivity analyses” can support such meetings by identifying which differences have little or no consequence on outcomes.

The five “Level 1” allocations for planning, scope, staffing, trainers, and facilities would cost \$115,000 altogether; the five “Level 5” allocations would cost an additional \$435,000 or \$550,000 altogether. From the entirely lowest to the entirely highest resource allocations, there are 3,125 possible combinations of investment levels (i.e.,  $5 \times 5 \times 5 \times 5 \times 5$ ). If all three groups shared the absolute value of reducing project costs, there would be no conflict with respect to the extent of preparation envisioned. Planning would be conducted as agenda points for currently scheduled meetings. The scope of preparation would involve one group of participants in a series of six one-day sessions. Staffing would be provided by the commitment of a senior clerical employee on half-time assignment. Trainers would be selected from current staff members. One of the regular meeting rooms in the central office would be reserved for instructional space; no food or beverages would be provided. These are all “Level 1” allocations that minimize project costs.

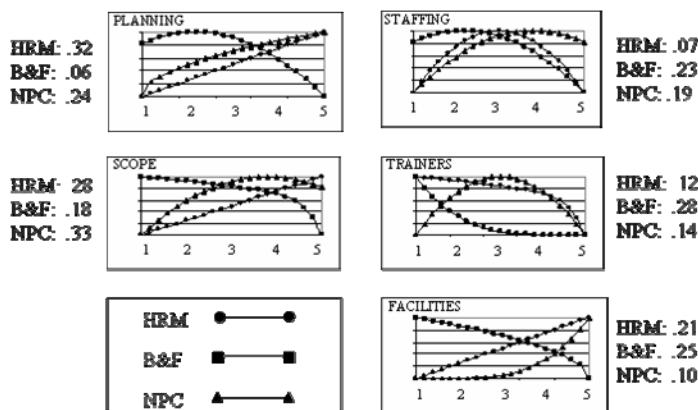


Fig. 2. Relative weights and function forms for three groups

	LEVEL 1	LEVEL 2	LEVEL 3	LEVEL 4	LEVEL 5
PLANNING	\$0 agenda pts. at meetings	\$4 half a day on-site	\$8 one day on-site	\$12 p.m. & a.m. off-site	\$23 two days off-site
SCOPE	6 1-day 1 group/15	6 1-day 1 group/25	4 2-day 2 groups/15	4 2-day 3 groups/15	4 3-day 2 groups/25
STAFFING	half-time sr. clerical	half-time manager	full-time manager	full-time mgr half-time clr	full-time mgr full-time clr
TRAINERS	in-house staffing	in-house w/ consultant	university team	regional contractor	national contractor
FACILITIES	\$0 in-house classroom	\$6 in-house w/ breaks	\$10 conference center	\$21 conf. ctr. w/ breaks	\$32 conf. ctr. w/ breaks & lunch

Fig. 3. Joint resource allocation structure with costs (in thousands)

The introduction of a second and competing value—increasing human capital—leads to the trade-offs being considered here. In an organizational preparation process

for a new and substantial initiative, enhancement of human capital is achieved with the expenditure of ever greater monetary amounts. The three teams from human resources management (HRM), budget and finance (B&F), and new project coordination (NPC) somewhat uniquely consider the relative importance of cost containment and human capital expansion. In this case, the application of VKM is critical to locating a specific proposal, expressed as one particular combination of investment levels out of the 3,125 possible, to which the three groups will agree and make a genuine commitment.

## 4 Application of Analytical Mediation

Analytical mediation is a computer-supported process used in conflictual situations to identify potential settlements with high joint benefits [15]. Integer goal programming provides a means for readily identifying settlements that lie on or near the efficient frontier. The basic objective for the application of analytical mediation is not to prescribe a specific settlement but, rather, in the spirit of the single-negotiating text idea proposed by Raiffa [16], to provide a concrete, externally authored proposal which the negotiating teams can criticize and use as a springboard for developing a settlement that might be considered as even more mutually satisfactory.

The use of analytical mediation for VKM in this case follows closely the method described by Mumpower and Rohrbaugh [17] and extended to multi-party resource allocation by Darling, Mumpower, Rohrbaugh, and Vari [18]. As illustrated in Figure 4, all possible settlements are arrayed in the joint utility space for each pair of teams. If a pair of teams share a similar commitment to preferred actions and goals, the points that are plotted appear around the diagonal from the lower left to the upper right (as shown for HRM and NPC). If a pair of teams differ in their commitment to preferred actions and goals due to opposing values, the points that are plotted appear around the diagonal from the upper left to the lower right (as shown for B&F and HRM, also for B&F and NPC).

Highlighted in Figure 4 are two regions containing 1) all settlements carrying a cost that is 25% of the total increased cost from minimum to maximum; and 2) all settlements carrying a cost that is 75% of the total increased cost from minimum to maximum. Many more such regions could be defined. Also identified are the points of minimum cost (all “Level 1” allocations) and of maximum cost (all “Level 5” allocations). The degree of overlap in the two regions—clearly visible for all three pairs of teams—indicates that considerable joint utility can be achieved without incurring large costs. In other words, the organization does not need to expend upwards to 75% of the total increased cost for the three groups to agree and make a genuine commitment to a shared organization preparation process; in fact, increased cost reduces the utility of settlements for the B&F group.

One proposed settlement identified in Figure 4 stands out in these graphs:

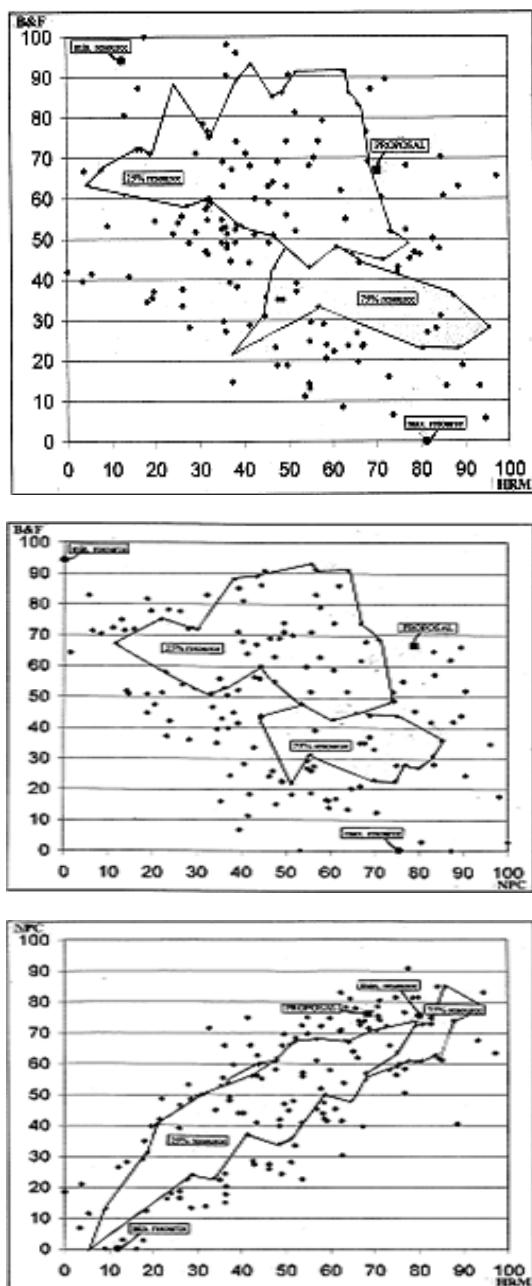
Planning Level 4: afternoon and following morning meeting off-site (\$12,000)

Scope Level 4: Three groups of 15 participants; four two-day sessions (\$115,000)

Staffing Level 3: full-time manager (\$90,000)

Trainer Level 2: in-house staff with consultant support (\$75,000)

Facilities Level 2: in-house space with light food and beverages during breaks (\$6,000)



**Fig. 4.** An illustration of analytical mediation

At a total cost of under \$300,000 (that is, about 40% of the total increased cost from minimum to maximum), this proposal provides between two-thirds and three-quarters of the total utility that would be gained by each group had their own “ideal”

plan of action been adopted.<sup>3</sup> On a utility scale from 0 to 100, this proposal provides the HRM group with 69, the B&F group with 67, and the NPC group with 77. Movement away from this proposal to other possible settlements appears to advantage one or two teams more greatly at the disadvantage of the other(s) but certainly is deserving of the groups' consideration.

## 5 Discussion and Conclusions

The present case—a decision about the allocation of resources to an organizational preparation process for a new and substantial initiative—offers a prime example of the importance of value knowledge management (VKM). Although value knowledge is an under-represented domain of study in the KM field, the effective articulation, codification, and communication of individual and group values remain highly consequential aspects of any organizational problem-solving or opportunity-seeking process. Since values, whether relative or absolute, are the standards which influence commitment to preferred actions and goals, an organization maintains its intentional course by acting in a value-coherent and value-correspondent manner [2].

Many organizational conflicts have integrative potential, that is, where the nature of the problem permits solutions that are better than zero-sum for all parties [19]; in such situations, each party can gain reasonably well and not necessarily at the expense of the others. Of course, the nature of the favorable “solution space” as depicted in Figure 4 would not be known without the application of VKM. In fact, the relative values of the three teams—HRM, B&F, and NPC—that undergird the plotting of joint utilities would not have been evoked explicitly without the use of the judgment analysis method in the initial VKM stage.

Even in organizational circumstances in which a single team is called upon to allocate resources, the challenge is made difficult because of the number of activities, projects, or programs that request (or require) support. Furthermore, experienced professionals realize that resource allocations rarely should be simplified as dichotomous choices (i.e., “go or no-go” choices between full investment versus non-investment); intermediate levels of resource commitment almost always exist and should be considered. In the present resource allocation model with merely five organizational actions being considered at only five levels of investment, the total number of alternative combinations exceeds 3,000, a highly complex task that increases geometrically with more actions and/or more levels.

When resource allocation decisions are shared by multiple groups bringing their own respective values to the process, the complexity of the task is made even greater. VKM provides an extraordinarily valuable approach for process structuring in multi-party conflict. The present trade-off between two key values—increasing human capital and reducing project costs—is considered from the unique perspective of each of the three teams. At a total cost of under \$300,000 (that is, about 40% of the total

---

<sup>3</sup> For HRM, the ideal would be levels 5, 5, 3, 1, and 5, respectively, at a cost of \$385,000. For B&F, the ideal would be levels 2, 1, 2, 1, and 1, respectively, at a cost of \$134,000. For NPC, the ideal would be levels 5, 4, 4, 3, and 5, respectively, at a cost of \$395,000. These levels can be identified directly from Figure 2 as the maximum points on each group's set of function forms.

increased cost from minimum to maximum), the proposal described in this case provides between two-thirds and three-quarters of the total utility that would be gained by each group had their own “ideal” plan of action been adopted. Arguably, without VKM substantial joint project gains and/or resource savings might be forfeited.

In conclusion, the importance of knowledge about individual and group values, as well as the management of such knowledge, should be an increasingly important domain of study within the KM field. This is especially true where the development of lateral relations and knowledge sharing across professional subgroups is of organizational interest [20, 21]. The present case illustrates one approach to VKM and demonstrates how the articulation of group judgment policies, the development of a shared resource allocation model, and the application of analytical mediation make a substantial contribution to organizational problem solving or opportunity seeking. The further development of VKM and the possibility of more frequent VKM applications should follow.

## References

1. Liyanage, C., Elhag, T., Ballal, T., Li, Q.: Knowledge communication and translation – a knowledge transfer model. *Journal of Knowledge Management* 13, 118–131 (2001)
2. Hammond, K.R.: Human judgment and social policy: Irreducible uncertainty, inevitable error, unavoidable injustice. Oxford University Press, Oxford (1996)
3. Easterby-Smith, M., Lyles, M.A.: The Blackwell handbook of organizational learning and knowledge management. Blackwell Publishing, Oxford (2003)
4. Scott, W.A.: Values and organizations. Rand McNally, Chicago (1965)
5. Alavi, M., Leidner, D.E.: Knowledge management and knowledge management systems: Conceptual foundations and research issues. *MIS Quarterly* 25, 107–136 (2001)
6. Keeney, R.L.: Value-focused thinking: A path to creative decision making. Harvard University Press, Cambridge (1992)
7. Cooksey, R.W.: Judgment analysis: Theory, methods, and applications. Academic, New York (1996)
8. Rohrbaugh, J.: The relationship between strategy and achievement as the basic unit of group functioning. In: Hammond, K.R., Stewart, T.R. (eds.) *The Essential Brunswik: Beginnings, Explications, Applications*. Oxford University Press, New York (2001)
9. Reagan-Cirincione, P.: Improving the accuracy of group judgment: A process intervention combining group facilitation, social judgment analysis, and information technology. *Organizational Behavior and Human Decision Processes* 58, 246–270 (1994)
10. Adelman, L.: Real-time computer support for decision analysis in a group setting: Another class of decision support systems. *Interfaces* 14, 75–83 (1984)
11. Carper, W.B., Bresnick, T.A.: Strategic planning conferences. *Business Horizons* 32, 34–40 (1981)
12. Phillips, L.D.: Systems for solutions. *Datamation Business*, 26–29 (April 1985)
13. Schuman, S.P., Rohrbaugh, J.: Decision conferencing for systems planning. *Information and Management* 21, 147–159 (1991)
14. Vari, A., Vecsenyi, J.: Experiences with decision conferencing in Hungary. *Interfaces* 22, 72–83 (1992)

15. Mumpower, J.L., Schuman, S.P., Zumbolo, A.: Analytical mediation: An application in collective bargaining. In: Lee, R.M., McCosh, A.M., Migliaresi, P. (eds.) *Organisational Decision Support Systems*. North-Holland, Amsterdam (1988)
16. Raiffa, H.: *The art and science of negotiation*. Harvard University, Cambridge (1982)
17. Mumpower, J.L., Rohrbaugh, J.: Negotiation and design: Supporting resource allocation decisions through analytical mediation. *Group Decision and Negotiation* 5, 385–409 (1996)
18. Darling, T.A., Mumpower, J.L., Rohrbaugh, J., Vari, A.: Negotiation support for multi-party resource allocation: Developing recommendations for decreasing transportation-related air pollution in Budapest. *Group Decision and Negotiation* 8, 51–75 (1999)
19. Walton, R.E., McKersie, R.B.: *A behavioral theory of labor negotiations*. McGraw-Hill, New York (1965)
20. Rangachari, P.: Knowledge sharing networks in professional complex systems. *Journal of Knowledge Management* 13, 132–145 (2009)
21. van der Spek, R., Kruizinga, E., Kleijzen, A.: Strengthening lateral relations in organisations through knowledge management. *Journal of Knowledge Management* 13, 3–12 (2009)

# Leveraging Organizational Knowledge Management through Corporate Portal

Kamla Ali Al-Busaidi

Information Systems Department, Sultan Qaboos University  
P.O. Box 20, PC 123, Al-Khod, Oman  
kamlaa@squ.edu.om

**Abstract.** This pilot study examines the role of corporate portal on leveraging organizational knowledge management (acquisition, conversion, application and protection). It also explores the business processes benefits (such as efficiency, effectiveness and innovation) and employees benefits (such as learning, adaptability and satisfaction) that result from supporting organizational KM through corporate portal. The preliminary analysis of instructors' utilization of corporate portal in an academic institution shows that providing tools through corporate portals to support knowledge conversion enhances the effectiveness and efficiency of business processes and employees' learning, whereas providing tools to support knowledge applications enhances the effectiveness of organizational processes as well as employees' learning, adaptability and satisfaction. Thus, the analysis indicates that knowledge conversion impacts business processes more than employees, whereas the knowledge application impacts employees more than business processes. Offering tools to support knowledge protection also improves the effectiveness of organizational processes. However, the preliminary analysis shows that knowledge acquisition process has no impact on business processes or employees.

**Keywords:** Knowledge management, KM processes, KM benefits, Corporate portal, Organizational knowledge management, Portals.

## 1 Introduction

Deploying information technologies (IT) tools that facilitate knowledge management and sharing are essential for the development of a knowledge-based economy. The utilization of IT tools improves the effectiveness and efficiency of the nations and organizations efforts to manage their knowledge, and build their human resources [26].

Portals are one of the IT tools that provide a common gateway into multiple distributed repositories. Portals provide an efficient access to relevant and accurate information and knowledge [23, 25]. There are several types of portals such as commercial portals, corporate portals, affinity portals, industry-wide portals, mobile portals etc [23, 25]. A corporate portal is a gateway into the organization's knowledge resources. Corporate portals improve employees' productivity by improving corporate information access [4]. As a knowledge management tool, corporate portals should provide tools that effectively support several knowledge management processes [7].

Corporate portal provides a single web-based entry to corporate information and knowledge located inside and outside the organization. Based on Aneja et al. (2000), a corporate portal includes internal and external information resources [4]. Internal information resources include internal websites, collaboration products, documents, organizational knowledge bases, and data warehouses; whereas, external information resources may include external websites, external content, news and news feeds, and external services.

The objective of the study is to examine the role of a corporate portal on leveraging organizational knowledge management (acquisition, conversion, application and protection) as identified by [14], and (2) to explore the business processes and people benefits that result from supporting organizational knowledge management processes through corporate portal.

There are number of studies that empirically investigated the effects of organizational KM process on organizational effectiveness such as those of [3, 8, 14, 18, 20]. However, these studies are not comprehensive on KM processes and benefits. There are limited studies that are focused on KMS users [19] and clear measurements of KMS users' satisfaction are still not well established [22]. Based on my knowledge, there are limited studies that investigated the specific benefits of each of the KM processes independently.

## 2 Background Literature

### 2.1 Knowledge Management Processes

Knowledge management systems are systems that manage knowledge throughout the organization; they are developed to assist individuals and organizations to store, retrieve, and transfer knowledge throughout the organization. Structured or unstructured explicit knowledge from internal or external sources can be stored in an Organizational KMS [12, 25].

Knowledge management is the management of organizational knowledge. Knowledge management processes have been classified in the literature in several dimensions, which are more or less the same. Gold et al (2001) indicated that organizational knowledge management capability is measured by providing tools and mechanisms that support four major knowledge management processes: knowledge acquisition, knowledge conversion, knowledge application and knowledge protection [14]. Davenport and Prusak (1998) classified KM processes as knowledge generation, knowledge codification and knowledge utilization[12]. Alavi and Leidner (2001) classified KM processes as knowledge creation, knowledge codification/storage, knowledge transfer, knowledge application[2], while Becerra-Fernandez and colleagues (2004) classified them as knowledge discovery, knowledge capture, knowledge sharing and knowledge application[6]. Several other frameworks of KM processes were summarized by [7, 16]. This study adopts Gold et al.'s (2001) classification to evaluate the KM processes as it has been highly tested in the KM research and it is more comprehensive than other classifications [14].

Corporate portal includes several features and tools that can support organizational KM processes: knowledge acquisition, knowledge conversion, knowledge application and knowledge protection. Corporate portal provides employees with a rich shared

information work space to create, exchange, store, retrieve, share and reuse knowledge; it has content space for information access and retrieval, communication space for conversation and negotiations, and coordination space for cooperative work tasks [13]. Additionally, a portal has a number of features including core capabilities, supportive capabilities and web services. Core capabilities of the portal include collaboration, integration, publication, search, personalization and taxonomy; whereas supportive capabilities include security, scalability and profiling [7]. Thus, corporate portal plays a major role on supporting organizational knowledge management for any organization.

## 2.2 Knowledge Management Benefits

The literature indicated that the use of KMS resulted in several individual and organizational benefits. Becerra-Fernandez et al. (2004) categorized knowledge management benefits as people benefits (learning, satisfaction, adaptability), organizational process benefits (efficiency, effectiveness, and innovation), products benefits (value-added products and knowledge-based products and organizational benefits (direct impacts such as return on investment and indirect impacts such as economies of scale and scope and sustainable competitive advantage) [6].

Alavi and Lidner (1999) found that the perceived benefits of KMS can be categorized as process outcomes and organization outcomes. Process outcomes include communication (enhanced communication, faster communication, more visible opinions of staff and increased staff participation); and efficiency (reduced problem solving time, shortening proposal times faster results, faster delivery to market, and greater overall efficiency). Organization outcomes include financial (increased sales, decreased cost and higher profitability); marketing (better service, customer focus, targeted marketing, proactive marketing); and general (consistent proposals to multi-national clients, improved project management and personnel reduction)[1].

Based on Herzberg's two factors theory, Hendriks argued that individuals share knowledge because of motivation factors rather than hygiene factors [17]. Motivation factors are related to achievement, responsibility, recognition, work-challenge, and operational autonomy. On the other hand, hygiene factors are salary, bonuses and penalties. KMS also improves individuals' performance and productivity in terms of time and speed of the knowledge sharing process [21].

Likewise, the deployment of corporate portal provides several benefits for organizations [15]. Corporate portal expands corporate reach, reduces operational cost, bolsters customer loyalty by eliminating delays, enhances online productivity through online tools, improves corporate competitiveness through effective web mechanisms, accelerates decision making through rapid access to relevant information and knowledge sources, and faster and reduces the cost of business processes.

## 3 Framework

### 3.1 Framework Development

The objective of this pilot study is (1) to examine the role of a corporate portal on leveraging organizational knowledge management and (2) to explore the benefits that

result from supporting organizational knowledge management processes through corporate portal. This study adopted Gold et al's (2001) KM processes( knowledge acquisition, knowledge conversion, knowledge application and knowledge protection) as it has been highly tested in the information systems(IS) research, and it is more comprehensive than other classification [14]. As for the benefits, the study adopted the Becerra-Fernandez and his colleagues' (2004) benefits classification of business processes (efficiency, effectiveness, and innovation) and people (learning, adaptability and satisfaction)[6].

KM improves organizational business processes such as marketing, manufacturing, accounting, engineering, and public relations. For academic institutions, KM also improves basic academic business processes such as consulting, education, research, publishing, and courses manufacturing [24]. KM improves business processes on three dimensions: Effectiveness (performing the most suitable processes and making the best possible decisions), efficiency (performing the processes quickly and at a low-cost) and innovation (performing the processes in a creative and novel manner that improves marketability) [6]. In addition, KM impacts employees in three dimensions: Learning, adaptability and job satisfaction. KM supports employees' learning through externalization, internalization, socialization, and communities of practice [6]. KM, also, improve employees' acceptance of change and their preparation to respond to change. Furthermore, KM offers employees with solutions to problems they face, which as a result improves their job satisfaction.

### 3.2 Knowledge Management Processes

**Knowledge Acquisition.** Knowledge acquisition process is the process of obtaining knowledge from internal and external sources. Several terms are used to describe this process such as acquire, seek, generate, create, capture and collaborate, and all these terms referred to knowledge accumulation [14]. Providing tools for knowledge generation and acquisition is important for the deployment of KMS as it creates an organizational knowledge repository for future organizational reuse. Knowledge capture and acquisition is essential for the establishment of organizational memory [6, 12]. With corporate portal, knowledge can be acquired from corporate information sources or collaborations between individuals as well as linkages between the organization and other alliances. Corporate portal provides a rich content space that enables searching, accessing and retrieving content from internal and external sources. It also includes collaboration and communication tools. These corporate portal tools speed up business processes by accessing relevant information and knowledge and eliminate delays [15, 25].

**Knowledge Conversion.** Knowledge conversion process is the process of making existing knowledge useful; it is the process of organizing, integrating, and combining, structuring, coordinating and distributing knowledge [14]. This KM process is critical because it standardizes organizational knowledge and makes it consistent and useful for utilization. It sets the stage for a successful knowledge application. This process improves the efficiency and effectiveness of the organizational knowledge. Two core capabilities of corporate portal that enables knowledge conversion are content integration and personalization. Corporate portal consolidates and synchronizes knowledge

from internal and external sources and provides a single personalized integrated view of the organizational intellectual capital [7].

**Knowledge Application.** Knowledge application is the process of actually using the knowledge to solve problems and make decisions. It includes the retrieval and application of knowledge. The main benefit of knowledge utilization and application for individuals is individual productivity, which is indicated by improvement on individuals' decision-making and innovation capabilities [3, 12, 20]. More specifically, productivity improvement means that individuals will improve their judgments and skills, which will help them, make better decisions and accomplish their work more efficiently. Knowledge application helped companies improve their efficiency and reduce costs [11]. The use of corporate portal may also improve the efficiency and the effectiveness of knowledge application. Corporate portal provides rich content that can be applied by users to solve problems and make decisions. Furthermore, corporate portal integrates internal information and knowledge sources, such as internal websites, collaboration products, documents, organizational knowledge bases, and data warehouses, with external information and knowledge sources, such as external websites, external content, news and news feeds, and external service [4].

**Knowledge Protection.** Knowledge protection process is related to the protection of the organization knowledge from illegal or inappropriate use. Protecting organizational knowledge provides a competitive advantage [14]. For achieving competitive advantage through organizational knowledge, knowledge should be rare and inimitable [5]. Securing organizational information is improving organizational efficiency and effectiveness, and its information quality [25]. One of the supportive capabilities of corporate portal is security [7]. Corporate portal protects corporate internal and external knowledge by including authentication tools such as users' names and passwords. In addition, corporate portal protects organizational knowledge by customizing and displaying corporate information and knowledge according to the users' authorization level.

## 4 Methodology

### 4.1 Sample

This pilot investigation includes only 25 participants who are academic staff in a public university in Oman; they are users of the university's corporate portal.

Based on the IT department, the objective of the university portal came from the need to have consolidated e-services for three types of users, students, faculty members, and other staff. The university portal is a dynamic web-based electronic gateway on the university's internal and external data resources. Information displayed is personalized, and designed to serve particular sectors of the campus community, that is different types of users. Pages accessed through the standard access authorization (username and password) issued to the university students, faculty, technical and administrative staff. Each type of user accesses the information and resources that he/she authorized to access.

The university portal has many features; some of these significant features are: Content management, resources aggregation, searching and indexing, personalization,

single sign-in and bi-lingual (English and Arabic interface and content). Thus, the university portal provides several functions that support and leverage the university's knowledge management.

About 80% of the participants were male, and all the participants have above average computer skills. About 48% had at least two years of portal-use experience, and only 16% of them had less than one year portal-use experience. The majority of the participants, 68%, were PhD holders; while 28% of them were MSc holders and 4% of them were BSc holders. About 24% of the participants were lecturers, 44% were assistant professor, 24% associate professors and 8% were full professors only 20% of participants with less than 2 years work experience.

## 4.2 Questionnaire

The questionnaire contained the investigated constructs for the quantitative analysis, along with demographic questions (e.g., gender, age, degree, portal usage experience, work experience, and job title). Construct measurements items were phrased according to a 5-point Likert scale (strongly disagree to strongly agree).

To make this assessment for this study, the questionnaire had 34 indicators that formed the independent constructs (KM processes) and dependent constructs (benefits); see Table 1. Constructs related to knowledge management processes were adopted based on [14], while the benefits were self-developed based on [6]. Check the Appendix for the measurements. The study was conducted in English (the typical medium of business activities in Oman).

# 5 Data Analysis and Results

## 5.1 Analysis Methodology

Data was analyzed by the SPSS 16. Preliminary analysis of this pilot investigation was based on the reliability, correlations and other standard statistical measures (such as means, maximums and minimums). Multi-variate analysis and hypothesis testing was not conducted due to the small sample size.

## 5.2 Constructs Reliability

The reliabilities of the measurements were evaluated through internal consistency reliability; the recommended level for internal consistency reliability is at least 0.70 [9]. Despite the small sample size, this preliminary investigation found that the reliability of the study constructs were high. Table 1 shows that the study constructs' reliability were all above 0.7 except for the knowledge acquisition, which is 0.623 (almost close to 7).

The mean values of the constructs shows that this investigated university portal provides almost average tools for knowledge acquisition, conversion and application, and above average for knowledge protection. The means also illustrates that the participants gave above average (greater than 3) for organizational processes benefits of effectiveness and efficiency, but below average for innovation. The participants also gave above average (greater than 3) for people benefits of satisfaction, learning and adaptability.

**Table 1.** Constructs Reliability

Construct	Total items	Reliability	Mean
Acquisition	3	0.62	2.82
Conversion	6	0.84	2.882
Application	6	0.80	2.89
Protection	4	0.92	3.24
Effectiveness	2	0.94	3.33
Efficiency	3	0.92	3.43
Innovation	2	0.93	2.50
Adaptability	2	0.77	3.06
Learning	3	0.91	3.06
Satisfaction	2	0.88	3.23

### 5.3 Correlations

Table 2 shows that the preliminary correlations analysis indicates that offering tools through corporate portal to support and leverage organizational knowledge management results in some benefits. First, offering tools to support knowledge conversion is significantly correlated with the efficiency of business processes (a correlation of 0.599 and a significance level of 0.01), the effectiveness of business processes (0.591; 0.01), and employees' learning (0.412; 0.05). Second, providing tools, through corporate portal, for knowledge application significantly correlated with the effectiveness of business processes (a correlation of 0.538 and a significance level of 0.01), employees' learning (0.623; 0.01), employees' adaptability (0.478; 0.05 and employees' job satisfaction (0.454; 0.05). Third, providing tools for knowledge protection only

**Table 2.** Constructs Correlations

Constructs	Effectiveness	Efficiency	Innovation	Learning	Adaptability	Satisfaction
Acquisition	.173	.183	.375	.343	.180	.170
Conversion	.599**	.591**	.246	.412*	.142	.341
Application	.538**	.391	.281	.623**	.478*	.454*
Protection	.461*	.392	.096	.219	.019	.244

\*\*. Correlation is significant at the 0.01 level (2-tailed).

\*. Correlation is significant at the 0.05 level (2-tailed).

significantly correlated with the effectiveness of business processes (a correlation of 0.461, and a significance level of 0.05) However, the correlations table, Table 2, shows that offering tools to support knowledge acquisition have no significant correlations with any of the business processes benefits or employees benefits.

## 6 Conclusions

Deploying information technologies (IT) tools that facilitate knowledge management and sharing are prerequisite for the development of a knowledge-based economy.

Portals are one of the IT tools that provide a common gateway into multiple distributed information and knowledge repositories. The objective of this pilot study was to examine the role of a corporate portal on leveraging organizational knowledge management, and to explore the benefits that result from supporting organizational knowledge management processes through corporate portal. Preliminary analysis based on 25 participants showed that providing tools through the corporate portals that support knowledge conversion impacted effectiveness and efficiency of organizational processes, whereas providing tools that support knowledge applications resulted in process benefits (effectiveness) as well as people benefits(learning, adaptability and satisfaction). Thus, the analysis indicated that the major impact of supporting organizational knowledge through corporate portal result from knowledge application process. In addition, the analysis indicated that knowledge conversion impacted business processes more than people, whereas knowledge application impacted people more than business processes. Offering tools that support knowledge protection also impacted organizational processes effectiveness. The preliminary analysis showed that knowledge acquisition had no impact on organizational processes or people. This could be traced to the low construct reliability.

Despite the small sample size, this pilot study provided practitioners and researchers with reliable measures that can be used to examine knowledge management processes and also reliable measures to examine the business processes and people benefits. This study's preliminary analysis also provided some insights for practitioners and researchers on the role of corporate portal on supporting and leveraging organizational knowledge management by assessing the impacts of using corporate portal for organizational knowledge management on business processes and employees. Future research may include larger sample size to conduct hypotheses testing and advanced regression analysis. Future research also may extend the measurements of the benefits to include products impacts and organizational impacts. Moreover, the inclusion of more users and more organizations will enhance the external validity and generalizability of the results.

## References

1. Alavi, M., Leidner, D.: Knowledge Management Systems: Issues, Challenges, and Benefits. *Communication of the AIS* 1(7), 2–37 (1999)
2. Alavi, M., Leidner, D.: Review: Knowledge Management and Knowledge Management Systems: Conceptual foundations and research issues. *MIS Quarterly* 25(1), 107–136 (2001)

3. Al-Busaidi, K. A.: A Socio-Technical Investigation of the Determinants of Knowledge Management Systems Usage. Unpublished doctoral dissertation: Claremont Graduate University, Claremont, CA (2005)
4. Aneja, A., Rowan, C., Brooksby, B.: Corporate portal framework for transforming content chaos on Intranets. *Intel. Technology Journal* Q1, 1–7 (2000)
5. Barney, J.: Firm Resources and Sustained Competitive Advantage. *Journal of Management* 17(1), 99–120 (1991)
6. Becerra-Fernandez, I., Gonzalez, A., Sabherwal, R.: *Knowledge Management*. Pearson Education Inc., New Jersey (2004)
7. Benbya, H., Passante, G., Belbaly, N.: Corporate Portal: A tool for knowledge management synchronization. *International Journal of Information Management* 24, 201–220 (2004)
8. Chang, S., Lee, M.: The Effects of Organizational Culture and Knowledge Management Mechanisms on Organizational Innovation: An empirical study in Taiwan. *The Business Review-Cambridge* 7(1), 295–301 (2007)
9. Chin, W.: The Partial Least Square Approach to Structural Equation Modelling. In: Marcolides, G.A. (ed.) *Modern Methods for Business Research*, pp. 295–336. Lawrence Erlbaum Associates, London (1998)
10. Chong, S.: KM Critical Success Factors: A comparison of perceived importance versus implementation in Malaysian ICT companies. *The Learning Organization* 13(3), 230–256 (2006)
11. Davenport, T., Klahr, P.: Managing Customer Support Knowledge. *California Management Review* 40(3), 195–208 (1998)
12. Davenport, T., Prusak, L.: *Working Knowledge*. Harvard Business School Press, Boston (1998)
13. Detlor, B.: The Corporate Portal as Information Infrastructure: Towards a framework for portal design. *International Journal of Information Management* 20, 91–101 (2000)
14. Gold, A.H., Malhotra, A., Segars, A.H.: Knowledge Management: An organizational capabilities perspective. *Journal of Management Information Systems* 18(1), 185–214 (2001)
15. Gurugé, A.: Living and Breathing Portals. In: *Corporate Portals Empowered with XML and Web Services*, pp. 273–284 (2002)
16. Heisig, P.: Harmonisation of Knowledge Management – Comparing 160 KM Frameworks around the Globe. *Journal of Knowledge Management* 13(4), 4–31 (2009)
17. Hendriks, P.: Why Share Knowledge? The influence of ICT on the motivation for knowledge sharing. *Knowledge and Process Management* 6(2), 91–100 (1999)
18. Jennex, M.: Impacts From Using Knowledge: A longitudinal study from a nuclear power plant. *International Journal of Knowledge Management* 4(1), 51–64 (2008)
19. Kankanhalli, A., Tan, B.: A Review of Metrics for Knowledge Management Systems and knowledge management initiatives. In: *The Proceedings of the 37th Hawaii International Conference on System Sciences*, Hawaii (2004)
20. Liu, S.: A Study of Factors that Facilitate Use of Knowledge Management Systems and the Impact of Use on Individual Learning. PhD Dissertation: Claremont Graduate University, Claremont, CA (2003)
21. Maier, R.: *Knowledge Management Systems: Information and communication technologies for knowledge management*. Springer, Berlin (2002)
22. Ong, C., Lai, J.: Measuring User Satisfaction with Knowledge Management Systems: Scale development, purification, and initial test. *Computers in Human Behavior* 23(3), 1329–1346 (2007)

23. Rainer, R., Turban, E., Potter, R.: *Introduction to Information Systems: Supporting and transforming business*. John Wiley & Sons Inc., USA (2007)
24. Tikhomirova, N., Gritsenko, A., Pechenkin, A.: University Approach to Knowledge Management. *VINE: The Journal of Information and Knowledge Management Systems* 38(1), 16–21 (2008)
25. Turban, E., Leidner, D., McLean, E., Wetherbe, J.: *Information Technology for Management: Transforming organizations in the digital economy*. Wiley & Sons, Inc., USA (2008)
26. World Bank, Technical Cooperation Program Brief on GCC (2003),  
<http://web.worldbank.org/WBSITE/EXTERNAL/COUNTRIES/MENAEXT/BAHRAINEXTN/0,,menuPK:312668~pagePK:141132~piPK:141107~theSitePK:312658,00.html>

## Appendix: Constructs Measurements

<b>Knowledge Acquisition Tools : Corporate portal has tools for:</b>
1.generating new knowledge(info) from existing knowledge
2. identifying best practices
3. acquiring new knowledge(information) from external sources
<b>Knowledge Conversion Tools: Corporate portal has tools for :</b>
1.transferring organizational knowledge(info) to individuals
2.distributing knowledge(information)throughout the organization
3.absorbing knowledge(info) from individuals into the organization
4.integrating different sources and types of knowledge(info)
5. converting competitive intelligence into plans of action
6. filtering knowledge(information)
<b>Knowledge Application Tools: Corporate portal has tools for :</b>
1.applying knowledge(information) learned from experiences
2. using knowledge(information) to solve new problems
3.locating and applying knowledge(information) to critical competitive needs
4. taking advantage of new knowledge(information)
5. matching sources of knowledge(information) to problems and challenges
6. making knowledge(info) accessible to those who needs it
<b>Knowledge Protection Tools: Corporate portal has tools for :</b>
1.protecting knowledge(information) from inappropriate use inside the organization
2. protecting knowledge(information) from inappropriate use outside the organization
3. restricting access to some sources of knowledge(information)
4.clearly communicating the importance of protecting knowledge(information)
<b>Processes Benefits: The use of corporate portal :</b>
<b>Effectiveness</b>
1. improves the effectiveness of my work
2. improves the quality of my work
<b>Efficiency</b>
1. improves the efficiency of my work
2. helps me complete my work quickly
3. helps me complete my work at lower cost
<b>Innovation</b>
1. improves my creativity at work
2. Improves my innovation at work
<b>People Benefits: The use of corporate portal :</b>
<b>Learning</b>
1. improves my learning process
2. enhances my personal knowledge
<b>Adaptability</b>
1. enhances my adaptability level at work
2. helps me adapt quickly to new tasks
3. helps me be responsive to new job demands
<b>Satisfaction</b>
1. enhances my job satisfaction
2. makes me more satisfied with my job

# Author Index

- Adrian, Benjamin 3  
Al-Busaidi, Kamla Ali 399  
Allocca, Carlo 164  
Aussenac-Gilles, Nathalie 237
- Bała, Piotr 377  
Bonniol, Stéphane 121  
Bruno, Eric 107  
Büchner, Thomas 351
- Camps, Valérie 237  
Cardillo, Elena 249
- d'Aquin, Mathieu 164  
David, Jérôme 210  
Dengel, Andreas 3  
Dounias, Georgios 81  
Duarte, F. Jorge F. 133  
Duarte, João M.M. 133
- Fang, Xing 35  
Fathi, Madjid 17  
Fred, Ana L.N. 133
- Gabadinho, Alexis 94  
Gniadek, Jolanta 337  
Goldstein-Stewart, Jade 276  
Guo, Weisen 53
- Hassan, Shahidul 390  
Herold, Axel 151  
Hicks, Amanda 151  
Hois, Joana 262  
Holland, Alexander 17
- Kivijärvi, Hannu 364  
Kraines, Steven B. 53
- Langer, Hagen 68  
Laurent, Anne 121  
Lavoué, Élise 310  
Li, Guofu 197  
Litz, Berenike 68
- Malaka, Rainer 68  
Marchand-Maillet, Stephane 107  
Matthes, Florian 351  
Memon, Nasrullah 337
- Mittal, Harsh 177  
Möller, Ralf 224  
Motta, Enrico 164  
Müller, Nicolas S. 94
- Neubert, Christian 351
- Piirainen, Kalle A. 364  
Poncelet, Pascal 121
- Reineking, Thomas 262  
Ritschard, Gilbert 94  
Roche, Mathieu 121  
Rodrigues, M. Fátima C. 133  
Rohrbaugh, John 390  
Rougemaille, Sylvain 237
- Sachdeva, Jitesh 177  
Saneifar, Hassan 121  
Saraiva, João de Sousa 323  
Scharffe, François 210  
Schult, Niclas 262  
Sellami, Zied 237  
Serafini, Luciano 249  
Singh, Jaspreet 177  
Silva, Alberto Rodrigues da 323  
Stocker, Alexander 297
- Studer, Matthias 94  
Šváb-Zamazal, Ondřej 210  
Svátek, Vojtěch 210  
Szekely, Eniko 107  
Szymański, Julian 187
- Tamilin, Andrei 249  
Tochtermann, Klaus 297  
Tsakonas, Athanasios 81  
Tuominen, Markku 364
- Veale, Tony 197
- Wandelt, Sebastian 224  
Wiil, Uffe Kock 337  
Winder, Ransom K. 276
- Zhan, Justin 35  
Zyglarski, Błażej 377