KNOWLEDGE WORK IN CONTEXT

USER CENTERED KNOWLEDGE WORKER SUPPORT

KNOWLEDGE WORK IN CONTEXT

USER CENTERED KNOWLEDGE WORKER SUPPORT

Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de rector magnificus,
volgens besluit van het college van decanen
in het openbaar te verdedigen op vrijdag 19 februari 2016, om 10.30 uur precies

door

Maya SAPPELLI

geboren op vrijdag 22 april 1988
te Eindhoven, Nederland.

Promotor:	Prof. dr. ir. W. Kraaij
Copromotor:	Dr. S. Verberne
Manuscriptcommissie:	Prof. dr. A.P.J. van den Bosch (voorzitter) Prof. dr. ir. A.P de Vries Prof. K. Järvelin (Tampereen teknillinen yliopisto)

This publication was supported by the Dutch national program COMMIT (project P7 SWELL)

SIKS Dissertation Series No. 2016-03 The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

ISBN 978-94-028-0009-8

An electronic version of this dissertation is available at `http://repository.ru.nl/`.

CONTENTS

DANKWOORD (ACKNOWLEDGEMENTS)

Voordat dit proefschrift voltooid was, is er in 4 jaar tijd veel gebeurd. Allereerst wil ik Wessel bedanken dat hij aan mij dacht voor dit project, en natuurlijk ook voor zijn begeleiding de afgelopen jaren. Soms was het wel eens lastig een gaatje te vinden in je volle agenda, maar het is altijd goed gekomen. Daarnaast heeft Suzan een grote rol gespeeld tijdens de totstandkoming van mijn proefschrift. Ik kon altijd terecht bij je met alle vragen en je was altijd snel met je (review)commentaar zodat ik ook vlot weer verder kon. Je was duidelijk over wat er beter kon, maar ook over wat er al goed was. Daarom vond ik de samenwerking met jou altijd erg prettig.

Natuurlijk zal ik ook Saskia niet vergeten; we begonnen (bijna) tegelijk en eindigen ook (bijna) tegelijk. Het was fijn om met iemand te kunnen sparren binnen hetzelfde project, maar ook om iemand te hebben bij wie ik kon spuien over deadlinestress en reviewcommentaar.

Maaike, je kwam wat later als PhD-student bij Wessel, maar ook met jou heb ik een leuke tijd beleefd. Het was leuk om met jou aan het Goose-project te werken, vooral omdat het even totaal niet aan SWELL gerelateerd was. En ook met jou kon ik lekker babbelen.

Ik wil Theo bedanken voor zijn hulp bij de formele kant van mijn thesis. Ik heb veel van je geleerd en het was erg leuk om samen te brainstormen over context en hoe we dit konden modelleren.

I want to thank Gabriella for the opportunity to come work in Milan. Even though I had to leave after only a week, I enjoyed my time with you and your group. I felt very welcomed by Ekaterina, Marco and Stefania. Moreover, I am glad that we were able to continue our cooperation when I got back home.

Ofcourse I also want to thank my committee: Antal, Arjen, Kalervo, Gabriella and Theo for taking the time to read and review my thesis.

Daarnaast zijn er tal van collega's die ik dankbaar ben voor hun hulp, maar vooral ook voor de gezelligheid de afgelopen jaren. Vooral Max moet ik bedanken voor de theepauzes, waarbij ik meestal te lang bleef praten. De sushiavonden met Max, Kasper, Barbara en overige wisselende samenstellingen (Josef, Carst, Alexandra, Simone, Daniel en wie ik nog vergeet) waren altijd een groot succes. Natuurlijk wil ook alle andere collega's van ICIS, het Information Foraging Lab en TNO bedanken voor de gezellige tijd. (Ik ga jullie niet allemaal bij naam noemen, want dat zijn er veel te veel!)

Ik wil verder speciaal wat mensen noemen die ik bij de onderzoeksschool SIKS heb ontmoet. Vooral met Chris, Laurens en Michiel heb ik veel lol gehad tijdens de pool-, bier-, en squashuitjes.

Ook mijn vriendinnen wil ik bedanken, vooral Linda voor haar hulp toen ik in tijdsnood zat. Maar ook Inge, Judith, Suzanne en Willemijn ben ik niet vergeten en verdienen een plekje in dit dankwoord. Ik ken jullie al vanaf mijn studie Taalwetenschap, en onze heerlijke etentjes zorgden altijd weer voor nieuwe energie! Natuurlijk leverde het jaarlijkse relaxdagje met Anouk ook altijd weer nieuwe energie op!

Mijn familie zal ik ook niet vergeten. Ik wil vooral mijn ouders en mijn broer Fedde bedanken. Zonder hen was ik niet de wetenschapper die ik vandaag de dag ben. Vooral onze eettafeldiscussies hebben mij geholpen om alles met een kritische blik te bekijken.

Als laatste zijn er dan nog twee hele speciale mensen in mijn leven. De eerste is Micha. Ook jij hebt me altijd gesteund tijdens mijn proefschrift. Je was altijd bereid mee te denken over de problemen waar ik tegenaan liep. Je hebt me heel concreet geholpen met wat SQL-queries. Je gaf me de tijd en ruimte om dit proefschrift te schrijven, ook al betekende het dat ik daardoor soms gestresst of moe was. Dan zorgde je altijd dat ik weer wat afstand nam zodat ik de volgende dag weer met frisse moed verder kon gaan!

De tweede is mijn zoontje Jurre. Tijdens de laatste maanden van mijn PhD-project groeide je al in mijn buik. Je was een enorme stok achter de deur om mijn proefschrift op tijd af te ronden. Uiteindelijk is dat maar ternauwernood gelukt: 3 dagen nadat ik mijn proefschrift ingeleverd had, diende jij je aan. Het was net alsof je er op gewacht had.

Maya Sappelli
Nijmegen, December 2015

1

INTRODUCTION

Stress at work is a problem that is increasing in its prevalence. It can lead to health problems such as burn-out, which has far reaching consequences for the employer. In the Dutch economy, the annual loss from sick leave due to excessive stress is approximately EUR 4 billion (Blatter et al., 2005).

In the project SWELL[1] we aim to address this problem by the development of ICT applications that minimize the risk of burn-out and improve the well-being of the employee. The focus of the project is on supporting knowledge workers, as they make up 25 percent of our workforce (Dankbaar and Vissers, 2009). A knowledge worker is a professional whose main job is to produce and distribute knowledge; to "think for a living" (Davenport, 2013). Examples are software engineers, researchers, librarians and lawyers.

In this thesis we aim to develop computational methods to improve the well-being of knowledge workers. We assume that a knowledge worker's well-being can be improved by making his life at work as easy as possible. One way to do so is by targeting factors that have a negative impact on his feeling of well-being at work. One of the largest negative impacts on well-being at work is information overload (Reuters, 1998).

We investigate two possible approaches to tackle information overload. The first is *personal information management (PIM)* (Bawden and Robinson, 2009). In this field the challenge is to develop algorithms that categorize the data of the knowledge worker in a meaningful way. This helps the user find and access his documents effectively and efficiently. Moreover, the use of these algorithms should require little user effort, in order not to further overload the knowledge worker further. For PIM we develop methods for e-mail categorization that require little user effort.

The second approach is *working in context* (Gomez-Perez et al., 2009; Warren, 2013). In this field the challenge is to develop algorithms that can detect the active context of the knowledge worker, and provide context-aware support. Again,

[1] http://www.swell-project.net : supported by the Dutch national program COMMIT (project P7 SWELL)

the use of these algorithms should require little user effort, such that the benefits for the knowledge worker are optimal. In this field we develop a method for context recognition and identification which can also be used for context-aware document recommendation.

For both domains a thorough understanding of the knowledge worker and his activities is important. There is, however, little evaluation data available for the development and evaluation of the types of algorithms proposed. This is a result of the fact that monitoring people implies accessing personal and company confidential data, which severely inhibits parties to make e.g. logging data available for research. We have made an attempt to accelerate research in this area, by collecting datasets of knowledge worker intents and computer interactions in a manner where privacy issues are controlled.

In the next section (1.1) we will describe a knowledge worker and the problems he faces during his day. The purpose is to give a good understanding of the situation of the knowledge worker. This is the point of departure for all choices and considerations that have been made in this thesis. It is followed by an explanation of the concepts of well-being at work and information overload (Section 1.2). They motivate the choices in technological solutions that we make to address the problems the knowledge worker faces at his desk. This chapter is concluded with the research questions and outline of the remainder of this thesis in Section 1.3 and a reading guide in Section 1.4.

1.1. THE KNOWLEDGE WORKER SCENARIO

In this section we will describe a persona: Bob. The scenario illustrates the problems a knowledge worker can face regarding information overload. These problems reduce the well-being of the persona.

Consider Bob, he is a 43 year old programmer at a large company. He starts his day with finishing up a report on his latest Java deep-learning project. Only a couple of details and citations are needed, but he needs to finish this work before 1 pm. He knows that the papers that he need as references in his report are somewhere on his computer, because he has read them before. At this point Bob could have been helped by opening these documents for him, as to spare him the time to navigate to them or look for them himself.

At 11 am he realizes that he is missing a piece of information. He has read it before, but cannot remember where and starts to search. Bob finds some information about deep-learning in Python. Because Python is relatively new to him, he finds it more interesting than his current Java project and he gets distracted. At 12.30 he realizes that he has spent to much time learning about deep-learning in Python and that he only has 30 minutes left to finish his project. He finishes it quickly. Bob could have been helped by making him aware of his distractions.

In the meantime a couple of e-mail messages have arrived for Bob. One is about the possibility to work on new, self-defined research. Bob has wanted this for a while, so decides to write a proposal. He already has an idea about the topic he wants to pursue, but he wants to challenge himself. At this point Bob could be helped by thinking out of the box, and suggesting him documents that are related to the topic,

but cover a variety of perspectives.

At 5 pm Bob finishes his day. He has found so many new documents for his new project proposal that he feels a little bit overwhelmed. He has not been able to read all documents yet, and there were also some documents that he used before and that he would like to re-read as they are relevant for the proposal as well. He decides to catch up on some reading at home. Moreover, there are still a couple of messages related to various projects that linger in his inbox. He was so busy with his work that he did not get around to reading or answering them. Bob feels a little unsatisfied about his progress today because there is so much work left over. He could be helped by prioritizing his unread e-mail messages so that he can process the messages efficiently.

1.2. WELL-BEING AT WORK

Bob has shown us that even when working hard, a knowledge worker can feel unsatisfied. The feeling of well-being is influenced by many aspects and not all of them are related to work. In order to understand how we can support Bob in his work we need to understand what constitutes to well-being at work. Maslach and Leiter (2008) state that "An individual's psychological relationships to their jobs have been conceptualized as a continuum between the negative experience of burnout and the positive experience of engagement." (p. 498). An engaged employee feels energetic, involved and has a high self-efficacy, while an employee on the brink of burnout feels exhausted, cynical and has a low self-efficacy.

An important model that explains the balance in the working life of a knowledge worker is the Job Demands–Resources model by Demerouti et al. (2001). This model assumes that when there is an imbalance between the job demands and the resources of the knowledge worker, the knowledge worker can feel strained, which can lead to exhaustion. On the other hand a good balance will make the knowledge worker feel engaged and energetic. This means that a high demanding job can, depending on the available resources, either give a feeling of well-being to the knowledge worker, or give stress to the knowledge worker. Job demands are "those physical, psychological, social, or organizational aspects of the job that require sustained physical or mental effort and are therefore associated with certain physiological costs (e.g. exhaustion)." (p. 501). Job resources are "those physical, psychological, social, or organizational aspects of the job that may do the following: (a) be functional in achieving work goals; (b) reduce job demands at the associated physiological and psychological cost; (c) stimulate personal growth, and development." (p. 501).

In the case of Bob we see that he has difficulty planning and finishing his activities in time. He tries to compensate by working in the evening, which only adds to his exhaustion. There are many ways in which Bob can potentially be helped. From the resource and demands model we can deduce many determinants that have a negative impact on well-being at work. One such determinant is stress. Stress is related to the imbalance in workload, disturbances and autonomy. In Section 1.2.1 we describe an important cause of stress in knowledge workers: information overload. Bob is also experiencing information overload in his work. Our work focuses on improving the balance between job demands and job resources using technological solutions that

1

are aimed at improving the aspects that help to achieve work goals. The technological solutions described in this thesis address information overload as a cause of stress at work. The proposed solutions are inspired by the suggestions described in Section 1.2.2

1.2.1. INFORMATION OVERLOAD

Information overload is recognized as an important influence on stress as well as job satisfaction (Reuters, 1998) This section describes research into the effects of information overload on well-being, as well as suggestions for reducing information overload.

In 1990, Schick, Gordon, and Haka (1990) defined information overload as "occurring when the information processing demands on an individual's time to perform interactions and internal calculations exceed the supply or capacity of time available for such processing". It is believed that this reduces an individual's decision making capabilities. The authors stated that information overload can be defined by the quantity of information that needs to be processed per unit of time. They believe that information overload can be reduced by making more efficient use of time (for example by standardizing operations, training or reducing the number of tasks that need to be performed) or increasing the time available in organizations (for example by increasing the time available for each individual or by expanding the workforce).

In addition, Ho and Tang (2001) defined two extra dimensions of the information overload problem. They consider information quantity (too much information), information quality (low quality of information) and the format (wide variety of formats) as the main determinants of information overload. The authors investigated this by reviewing statistical reports on information overload within five industry cases. They found that most solutions for the information overload problem were centred on the control of information quantity, such as Schick, Gordon, and Haka (1990) propose. However, they think that an effective solution should address all three dimensions (See Figure 1.1).

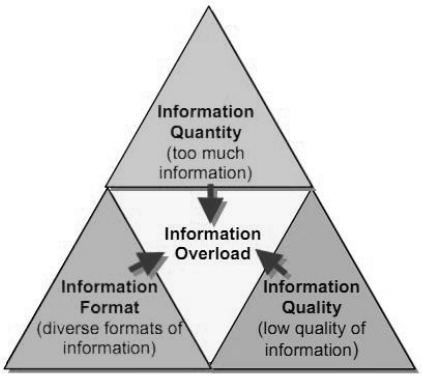


Figure 1.1: Dimensions in the information overload problem (Ho and Tang, 2001)

Ruff (2002) broadened the research on information overload with a review study on the effects of information overload on performance, physical health and social relations. They summarize several studies in which more than 60% of the employees reported a negative impact of information overload on personal and collegial relationships. Additionally, more than 25% of workers experience stress or health issues caused by information overload. Problems with concentration, multi-tasking, hurry sickness (feeling of constant rush), over stimulation (trance-like state), compulsion to check mail and internet to stay in touch, stress and burnout experiences are reported as effects of information overload. As a means to prevent information overload they suggest among others the use of a personal information system for storing and retrieving information, time-management and the "waste-not want-not mentality" (throw away unnecessary information). When information overload has already occurred they suggest among others filtering (focusing attention on the most important information), escaping (limiting disruptions from the outside world), prioritizing tasks, satisficing (use "good-enough" rather than "perfect" solutions) and limiting (accepting that more information is not always better). The technological solutions to support knowledge workers that we investigate are inspired by Ruff's suggestions to use a personal information system to deal with information overload, and to filter information to prevent information overload. In addition the technological solutions in this thesis are aimed at keeping the effort for the knowledge worker to use the supportive technology as low as possible.

One important source of information in the knowledge worker environment is e-mail. Whittaker and Sidner (1996) investigated the concept of e-mail overload. This form of information overload is believed to be caused by the misuse of the original purpose of the e-mail system. The authors stated that although e-mail was originally developed for the purpose of asynchronous communication, it is being used for task management, scheduling and personal archiving as well. This causes cluttered inboxes and information getting lost in archives. Their conclusions were based on semi-structured interviews with 20 users and the quantitative analysis of the mailboxes of these users.

Spira and Goldes (2007) add that there is a responsibility for the sender of a message when it comes to preventing e-mail overload. This means that you should think about which recipients are necessary and only send a message to those who need it. Furthermore, e-mails should be to the point and about only one topic at a time, with clear formulation and a subject that is clear and reflects the content of the mail.

The assumption that e-mail is indeed a large source of information overload is confirmed by Gantz, Boyd, and Dowling (2009). They measured information overload in an internet-based survey among 500 information workers in the US. They found that aspects of information overload such as the time it takes to reformat information, to search for but not find information and to recreate content, sum up to almost a full working day a week. 50% of the information that causes information overload originates from e-mail. The survey respondents are said to spend 26% of their time to manage information overload. Since e-mail is such an important cause of information overload, we investigate techniques that can be used for technological solutions that improve the organization of e-mail messages (Chapters 7 and 8).

In more recent work, the definition of information overload is refined to the idea that information overload occurs when the received information becomes a hindrance rather than a help, even though the information is potentially useful (Bawden and Robinson, 2009). According to Bawden and Robinson (2009), the reason that e-mail is often regarded as the worst offender when it comes to overload, is that its active delivery system is out of the user's control. Effects of such overload include (a) information anxiety: a feeling of stress caused by the inability to access, understand or make use of necessary information, (b) infobesity: a situation of personal information overload which often results in information avoidance where relevant information is ignored because there is too much to deal with, and (c) satisficing coping strategy: a situation where a coping strategy is used in which just enough information is retrieved to meet an information need with the risk of missing information. The authors state that the solution to information overload revolve around taking control of one's information environment, for example by better information management. The techniques for e-mail categorization investigated in this thesis are a form of information management.

So far, the presented research was concerned with information overload in a digital environment. Another source of overload could come from the workplace. Misra and Stokols (2011) investigated the effects of perceived information overload in cyber-based sources of overload, such as the internet and cellphones, and place-based sources of overload originating from physical settings, such as noise and crowding. They found that perceived cyber-based overload caused stress, while this was not the case for place-based overload. This suggests that digital sources of information play a larger role in overload than physical sources of information, which is a motivation to focus on digital sources and digital solutions.

A recent study by Benselin and Ragsdell (2015) suggests that there is a difference in perceived overload between younger and older people. Older people are less dependent on technology making them feel less overloaded. The cause of feeling overloaded in younger people came from lower levels information literacy. Information literacy can be defined as the ability to manage information, to use information and to absorb or remember information. The authors suggest that searching and managing information may help them. In Chapter 9 we investigate information recommendation as a method to support feelings of overload due to low information literacy.

1.2.2. APPROACHES TO REDUCE INFORMATION OVERLOAD

In this thesis we investigate two main lines of solutions that are aimed at reducing the information overload of the knowledge worker and thereby improving his well-being during his workday. The first is *personal information management (PIM)*: organizing information, a solution suggested by Bawden and Robinson (2009). In this field we target the organization of e-mail messages. The second solution is *context-aware support* during the workday, which is aimed at achieving work goals more efficiently and effectively, without increasing information overload. An example of context-aware support is pro-active information delivery (information recommendation). In this section we summarize the state of the art related to these two lines of knowledge worker support from the perspective of information science, information retrieval

and recommender systems.

PERSONAL INFORMATION MANAGEMENT

One solution to address information overload is personal information management (PIM). In order to understand what we can improve in PIM we look at literature from a psychological perspective by Lansdale (1988). He notices that a lot of people find it difficult to manage their information which can result in "messy" desks. On the one hand he thinks this is because of personal style, on the other hand that it is caused by the requirement of jobs to be flexible to new information demands. He found that people sometimes use "mess" to serve as a reminder for action.

Furthermore he notices a general problem with categorizing items. It is difficult to determine a categorization for items, but it is also difficult to remember what labels were used for the categorization. This is because most information items do not fall into neat categorization structures and category names can be ambiguous.

The fact that categorization functionality in e-mail clients is not often used optimally seems to confirm the analysis of Lansdale (1988). Many e-mail clients have an option to categorize, label or folder messages, but still, messages are left to linger in the inbox (Whittaker and Sidner, 1996). Many users do not even use category folders at all (Koren et al., 2011; Grbovic et al., 2014). Manually categorizing the messages immediately simply takes too much time, diminishing the actual benefits of the categorization. Additionally, searching for e-mails has become the norm. This illustrates that the trade-off between e-mail categorization and user effort is not balanced, which is why the functionality is not adopted by users, even though it is often suggested as a solution to overload.

Automated approaches for e-mail message classification are plentiful. The early work in e-mail classification was mostly directed towards detecting spam (Sahami et al., 1998). This was followed by work towards categorizing e-mails in order to support PIM (Segal and Kephart, 1999; Bekkerman, 2004). Now, work on classifying e-mails has shifted from topical categorization to prediction of priority (Dredze et al., 2008; Aberdeen, Pacovsky, and Slater, 2010). However, understanding e-mail and interactions with e-mail in order to better support people is gaining interest again (Hanrahan, Pérez-Quiñones, and Martin, 2014; Kalman and Ravid, 2015).

Many of these automated approaches require training examples to operate properly. Acquiring these training examples takes effort and time from the user, something that a knowledge worker is limited in already. Therefore we aim for developing methods for automatic e-mail categorizations in this thesis that require little or no effort from the user, but that are still meaningful and beneficial to the user.

WORKING IN CONTEXT

Another way to support a knowledge worker and to give the knowledge worker control over his information situation is to use his context. Research indicates that people remember much about the context of a document (Blanc-Brude and Scapin, 2007; Kelly et al., 2008; Chen and Jones, 2014). For example, a person remembers where he was sitting while writing the document, or what the document looked liked when it was finished. Sometimes, the context is remembered better than the exact content of a document.

It is believed that "working in context" can be an efficient way to support the knowledge worker in his work life (Gomez-Perez et al., 2009; Warren, 2013). The idea is that context, and more specifically the task context of knowledge workers can be used to help the user stay focused on his tasks. The task context can help in gaining a more specific understanding about which documents are relevant at a certain time.

Maus (2001) have interpreted "working in context" as work flow support and describe support systems called Workflow Management Systems. These systems model tasks, business rules, users and applications in order to automate business processes. In later work, Maus et al. (2011) present an approach named ConTask. This system strives for a task-centric structure of the personal knowledge space. It works in a semantic desktop environment where semantic annotations and relations are made between information objects. Tasks can be defined, which offer the possibility for pro-active information delivery.

An approach that is not dependent on semantic annotations of information objects is described by Gomez-Perez et al. (2009). In their project ACTIVE, context is used to support information delivery and sharing. They aim to partition a set of information objects into contexts, either manually or using machine learning. The resulting contexts can be used for pro-active information delivery. The use of context as a tool to prevent information overload and to share information was validated in a case study described in Warren (2013).

Another approach to "working in context" is described in Biedert, Schwarz, and Roth-Berghofer (2008) where the knowledge worker is supported using a context-sensitive dashboard. In this system, applications and information objects are remembered and stored with a certain context. When the user switches context, all objects previously stored with that context are accessed. This can save the user time in navigating to the individual objects.

The downside of these approaches is that the context of the knowledge worker is limited to the (categorization of the) tasks of the knowledge worker. In the presented systems the knowledge worker himself defines tasks that he is interested in by recording his desktop activities or annotating information objects. There is no automatic discovery of tasks. The manual process of defining tasks requires effort from the user, which decreases the benefits of the system.

In this thesis we will investigate context-sensitive information delivery systems similar to the ACTIVE approach. Our approach, however, will integrate both context detection and context-aware support. Furthermore, the context that is used is not limited to fixed categorizations, but is dynamic and flexible. Where task or context categorisations are deemed useful, we will ensure that these can be defined with little user effort.

1.3. RESEARCH QUESTIONS AND THESIS OUTLINE

The focus of this thesis is on the development of computational methods for understanding the knowledge worker's context and using that context efficiently and effectively for supporting his daily activities. The assumption is that this targets the information overload that the knowledge worker experiences in his work. By targeting information overload, we assume that we can improve the knowledge worker's

well-being. In contrast to the research described in Section 1.2.2.2 we take a holistic, user centred approach where we evaluate our methods on realistic and noisy data. The quality of the methods is quantified by measuring effectiveness and user effort. These are aspects that are important in the knowledge worker setting.

We formulate the following main question to be answered in this thesis.

Main RQ How can we design, implement and evaluate context-aware methods that make computer-based knowledge work more effective and more efficient?

In order to answer this question we need to address three main points. First we need to understand more about the knowledge worker and his tasks. Second, we need to properly define a computational model to recognize the context that is needed for the context-awareness. And finally we need to provide context-aware support methods that can deal with the limitations of knowledge workers and their observed data.

1.3.1. PART 1: UNDERSTANDING THE KNOWLEDGE WORKER

In part 1 of this thesis we investigate the tasks and behaviour of the knowledge worker in a work setting. In Chapter 2 we look at how he represents his information need through queries during search activities. In Chapter 3 we look at how he conveys and interprets task in e-mail messages. In Chapter 4 we look at how the knowledge worker interacts with his computer while executing typical knowledge worker tasks such as writing reports and preparing presentations. These chapters are centred around the question:

RQ 1. What information about knowledge worker intent can we observe from interactions with the computer and what information do we need to deduce from other sources?

The challenge in this question is that there are no public datasets available on knowledge worker interactions with their computers. The main reason is that this information is often privacy sensitive, so when they are collected they are usually not shared with the research community.

Furthermore, the data itself is often challenging as computer interaction data contains much noise. This makes it difficult to find the intentions of the user in the data. We hypothesize that some information about tasks can be derived directly from the data trace that originates from queries and e-mail messages. However, other task information is implicit and can only be understood through associations that the knowledge worker makes, or by making connections to other computer interactions.

The main contributions in this part of the thesis are three labelled datasets for the research community with preliminary analyses that give insight into the activities of a knowledge worker during his work.

1.3.2. PART 2: CONTEXT OF THE KNOWLEDGE WORKER

In part 2 of this thesis we describe the conceptual and formal model of the knowledge worker context (Chapter 5) based on the lessons that we learned in part 1. Additionally, we describe an implementation of the model which provides the automatic

detection (Chapter 6) of the knowledge worker context. These chapters are centred around the following questions:

RQ 2. How should we define the context that we need for context-aware support for knowledge workers?

RQ 3. How can our conceptual model be implemented and how well can it detect the active context?

The main challenge in these problems is that the context of the knowledge worker is dynamic and there are many factors that influence this context.

We hypothesize that the knowledge worker's context consists of a complex combination of information, such as topics, entities, but also the location of the person and elements in that location. Even emotions and the time of day can play a role in the knowledge worker's context. In order to effectively support the knowledge worker, the context detection algorithm should be able to integrate various sources of information. Furthermore it should be capable of dealing with the dynamic nature of the context, such as switches between applications that can be made in a very short time span.

The main contribution of this part of the thesis is an integrated and dynamic algorithm for context detection in a knowledge worker support setting that requires little training effort from the user. Although we evaluate the model in the area of context-aware information support and PIM, its design supports the application in other domains as well. Future work includes the application of the model in predicting stress and emotion.

1.3.3. PART 3: CONTEXT-AWARE SUPPORT

As a last part of this thesis we look into various applications for knowledge worker support. In Chapter 7 and 8 we consider methods to support information management in e-mail applications, such as a categorization based on whether a user needs to reply to a message, or a categorization based on tasks for the user. These are focused around the question

RQ 4. How can we reduce user effort in training algorithms for e-mail categorization?

The challenge in this question is the reduction of user effort while ensuring the meaningfulness of the categories. The reduction of user effort is often ignored in the evaluation of classification methods. Especially in the knowledge worker scenario, where the user is often overloaded, the balance between user effort and benefits of the categorization is important. This has a large influence on whether the technology will be adopted. Without the adoption of the technology, the knowledge worker is not supported.

We believe that generic supervised categorization methods are often not sufficiently efficient in the user effort they require. Moreover unsupervised methods require little user effort, but their categorizations are often not meaningful enough. We

hypothesize that by making use of existing categorizations such as foldered documents, and meta data from e-mail history the user effort required for categorization can be diminished while the meaningfulness of the categories is kept intact.

The first contribution is a simple and transparent method for the prediction whether a message will be replied to, which is an aspect of the priority of an e-mail message. The second contribution is a new machine learning algorithm for e-mail categorization that leverages pre-existing labelled information sources and reduces the user effort required for training the method.

In Chapter 9 we describe context-aware document recommendation and its evaluation in order to answer the following question:

RQ 5. How should we evaluate context-aware information recommendation and what are the benefits and downsides of various methods for context-aware information support?

The challenge in this question is that typically the focus of recommender system evaluation is on the effectiveness of the algorithm. In the more complex and holistic knowledge worker scenario, however, effectiveness is not the only factor that determines whether the user benefits from the recommendations or not.

We hypothesize that a single evaluation criterion does not suffice when evaluating for knowledge worker support. Rather a combination of evaluation criteria should be used to determine whether a system will actually benefit a knowledge worker. The main contribution is a new integrated holistic approach to context-aware document recommendation and a description of a multi-faceted evaluation of context-aware document recommendation systems from the perspective of a knowledge worker.

Before we end this thesis with a conclusion that answers these questions, we present a chapter on the context-aware recommendation of tourist sights. This is an application that is not centred around the knowledge worker or his well-being at work and therefore is not associated with a formal research question. However, the chapter illustrates some interesting additional possibilities for context-aware recommendation systems. For example, the possibility to leverage both personal preferences and the preference of the masses or the possibility to influence people by describing the tourist places using positive reviews.

1.4. GUIDE FOR THE READER

This thesis is a collection of published papers and papers that are submitted for publication. Each chapter is a paper that can be read independently from the other chapters. This means that when reading the thesis as a whole, some repetition is unavoidable. However, since most chapters cover different topics the redundancy is not large. Thus, in order to cater the readers that will not read the entire thesis, we have chosen to edit the chapters as little as possible.

I

UNDERSTANDING THE KNOWLEDGE WORKER

While physics and mathematics may tell us how the universe began, they are not much use in predicting human behavior because there are far too many equations to solve. I'm no better than anyone else at understanding what makes people tick, particularly women.

Stephen Hawking

2

COLLECTING GROUND TRUTH DATA FOR QUERY INTENT

Edited from: **Maya Sappelli, Suzan Verberne, Maarten van der Heijden, Max Hinne, Wessel Kraaij** (2012) *Collecting ground truth data for query intent.* In: Proceedings of the Dutch-Belgium Information Retrieval workshop (DIR 2012)

Search engines try to support people in finding information and locating services on the web. What people are looking for depends on their underlying intent and is described by the query they enter in the search engine. These queries are often short and ambiguous. This chapter describes the collection of ground truth data for query intent. Participants were asked to label their own search queries according to what they hoped to find with that query. The data can be used to investigate the reliability of external human assessors and to train automatic classification models.

2.1. INTRODUCTION

All popular web search engines are designed for keyword queries. Although entering a few keywords is less natural than phrasing a full question, it is an efficient way of finding information and users have become used to formulating concise queries. For example, in the query log data set "Accelerating Search in Academic Research Spring 2006 Data Asset" released by Microsoft, 70% of the 12 Million queries (which were entered into the MSN Live search engine) consist of one or two words.

It seems unlikely that a few keywords can precisely describe what information a user desires, which we refer to as *search intent* (also known as *query intent*).[1] The exact definition of this concept is still a topic of debate (Gayo-Avello, 2009; Silvestri, 2010); but we can say that, roughly, search intent is what the user implicitly hoped to find using the submitted query. This is different from Broder's definition of information need (Broder, 2002) in that information need can be defined as the drive to

[1] We use the terms 'query intent' and 'search intent' interchangeably.

formulate a series of queries. The intent of a specific query is often part of a bigger information need. This is the case when only part of the information need is expected to be satisfied by a query, for example because the information need is too big to be expressed in a single query. If the intent behind a query is known, a search engine can improve on retrieval results by adapting the presented results based on the more specific intent instead of the (underspecified) query (White, Bennett, and Dumais, 2010).

Several studies have proposed classification schemes for query intent. After studying a large collection of AltaVista query logs, Broder (2002) suggested that the intent of a query can be either informational, navigational or transactional. Later, many expansions and alternative schemes have been proposed, which we will summarize in Section 2.2. Ultimately, a search engine should be able to automatically classify a query according to such a scheme, so that the search intent of the user can be taken into account in the retrieval result. However, for the implementation of automatic classification models, training data is needed: a set of queries, labelled with their underlying intent. In previous studies, annotations of query intent labelling have been created by human assessors (Baeza-Yates, Calderón-Benavides, and González-Caro, 2006; González-Caro et al., 2011). However, in those studies, the assessors are not the searchers themselves.

We asked search engine users to label their own queries according to the underlying intent. This provides a ground truth that can be used (1) to investigate the reliability of external human assessors and (2) to train automatic classification models. Our data set is an important contribution to the field of query intent classification, since many studies rely on classifications by external assessors as gold standard classifications because they do not have access to classifications by the searchers themselves. We intend to make our data set publicly available.

Our chapter is structured as follows. In Section 2.2 we describe intent classification schemes from the literature; Section 2.3 presents our classification scheme. In Section 2.4 we describe the user study that we conducted. The results of these experiments are presented in Section 2.5. Lastly, Section 2.6 concludes our work and described future research that we plan with this data.

2.2. INTENT CLASSIFICATION SCHEMES IN THE LITERATURE

The early paper by Broder (2002) presents a taxonomy of web search, defining three categories for the intent behind queries: navigational (the user wants to reach a particular website), informational (the user wants to find a piece of information on the web) and transactional (the user wants to perform a web-mediated task).Rose and Levinson (2004) refine the intent classification by Broder. They define three main categories for query intent: navigational, informational (which consists of five subcategories: directed, undirected, advice, locate, list) and resource (download, entertainment, interact, obtain).

More recently, it has been argued that search intent has more dimensions than the navigational – informational – transactional classification by Broder. Baeza-Yates, Calderón-Benavides, and González-Caro (2006) present a classification scheme with two dimensions: topic (categories taken from the Open Directory

Project[2] and goal (informational, non-informational or ambiguous). Sushmita, Piwowarski, and Lalmas (2010) distinguish between "query domain" (e.g. image, video, or map) and "query genre" (e.g. news, blog, or Wikipedia).

Calderón-Benavides, González-Caro, and Baeza-Yates (2010) and González-Caro et al. (2011) present multiple dimensions of user intent. Some of these are very general, such as Genre, Topic and Task (informational or non-informational). Others are better defined, such as Specificity and Authority sensitivity. The authors manually classify 5,000 queries according to all dimensions and give a good analysis of the agreement between judges and the correlation between dimensions.

Hinne et al. (2011) propose an intent classification scheme with three dimensions: topic, action type and modus.

2.3. OUR INTENT CLASSIFICATION SCHEME

We introduce a multi-dimensional classification scheme of query intent that is inspired by and uses aspects from Broder (2002), Baeza-Yates, Calderón-Benavides, and González-Caro (2006), González-Caro et al. (2011), Sushmita, Piwowarski, and Lalmas (2010) and Hinne et al. (2011) (see Section 2.2). Our classification scheme consists of the following dimensions of search intent.

1. Topic: categorical, fixed set of categories (from ODP).
2. Action type: categorical, consisting of:
    - informational,
    - navigational,
    - transactional.
3. Modus: categorical, consisting of:
    - image,
    - video,
    - map,
    - text,
    - other.
4. source authority sensitivity: 4-point ordinal scale (high sensitivity: relevance depends more on authority of source).
5. location sensitivity: 4-point ordinal scale (high sensitivity: relevance depends more on location).
6. time sensitivity: 4-point ordinal scale (high sensitivity: relevance depends more on time).
7. specificity: 4-point ordinal scale (high specificity: very directed goal; low specificity: explorative goal).

The topic should give a general idea of what the query is about, for which we use the well-known Open Directory Project categories. *Action type* is the Broder categorisation; Modus is based on Sushmita, Piwowarski, and Lalmas (2010). The ordinal di-

[2] Open Directory Project (ODP): http://dmoz.org

mensions are inspired by González-Caro et al. (2011). While many more dimensions can be imagined, we think that these capture an important portion of query intent.

2.4. DATA COLLECTION

Table 2.1: Explanation of the intent dimensions for the participants.

Dimension	Explanation
Topic	What is the general topic of your query?
Action type	Is the goal of your query: (a) to find information (informational), (b) to perform an on-line task such as buying, booking or filling in a form (transactional), (c) to navigate to a specific website (navigational)?
Modus	In which form would you like the intended result to have?
Source authority sensitivity	How important is it that the intended result of your query is trustworthy?
Location sensitivity	Are you looking for something in a specific geographic location?
Time sensitivity	Are you looking for something that is related to a specific moment in time?
Specificity	Are you looking for one specific fact (high specificity) or general information (low specificity)?

In order to obtain query labels from search engine users, we created a plug-in for the Mozilla Firefox web browser. After installation by the user, the plug-in (locally) logs all queries submitted to Google and other Google domains, such as Google images. We asked colleagues (all academic scientists and PhD students) to participate in our experiment. Participants were asked to occasionally annotate the queries they submitted in the last 48 hours, using a form showing our intent classification scheme. Table 2.1 shows the explanations of the intent dimensions that were given to the participants. To ensure participants understood what they were asked to do, we first presented three reference queries which were the same for all participants. Other queries were displayed in chronological order.

In order to avoid privacy issues, participants were allowed to skip any query they did not want to submit. When a participant clicked the 'submit' button, he was presented with a summary of his queries, from which queries could be excluded once again. After confirmation, the queries and annotations were sent to our server. For each submitted query, we stored the query itself, a time stamp of the moment the query was issued, a participant ID (a randomly initiated number used to group queries in sessions per participant) and the annotation labels.

2.5. RESULTS

In total, 11 participants enrolled in the experiment. Together, they annotated 605 queries with their query intent, of which 135 were annotated more than once[3]. On

[3]It is important to notice that it is possible to represent different search intents with the same query, for example, when the same query is issued by the same person at different times.

(a) Number of queries per action type

Action type	# Queries[a]
Informational	546
Transactional	30
Navigational	70

[a]The sum of the queries may be higher than 605 since multiple action types could be selected per query

(b) Number of queries per modus

Modus	# Queries[a]
Image	33
Video	10
Map	27
Text	512
Other	6

[a]The sum of the queries may not add up to 605 since multiple modi could be selected per query and modus could be omitted

(c) The most frequently selected categories

Category	# Queries
Computers	250
Science	193
Recreation	87
Health	84
Reference	76

Table 2.2: Number of queries for the various dimensions

average, each person annotated 55 queries. Table 2.2(a) shows the number of queries per action type as annotated by the participants. Table 2.2(b) shows the number of queries per modus as annotated by the participants.

The three topic categories that were used most frequently in the annotated queries were *computer*, *science* and *recreation*. Table 2.2(c) shows the five most frequent categories and their frequencies. Figure 2.1 displays the labelling distributions, from low to high, for the following ordinal dimensions: source authority sensitivity, location sensitivity, temporal sensitivity and specificity of the queries.

2.5.1. ANALYSIS

In this section we take a closer look at the annotated queries. We first calculated correlations between the classification dimensions. The correlation between ordinal dimensions (source authority sensitivity, location sensitivity, temporal sensitivity and specificity) was estimated using Kendall's τ-b measure. We found a significant moderately strong positive correlation between source authority sensitivity and specificity ($\tau = 0.377$, $p < 0.0001$) as well as between location sensitivity and temporal sensitivity ($\tau = 0.421$, $p < 0.0001$). Additionally, we found weak positive correlations between location sensitivity and source authority sensitivity ($\tau = 0.135$, $p = 0.0002$) and between location sensitivity and specificity ($\tau = 0.106$, $p = 0.0042$).

Correlations for the categorical dimensions (topic, action type and modus) were determined using a chi-squared test. However, the outcome of this test is unreliable because there were too many zero-occurrences in the cross table of the dimensions. We do, however, see some interesting trends in the data:

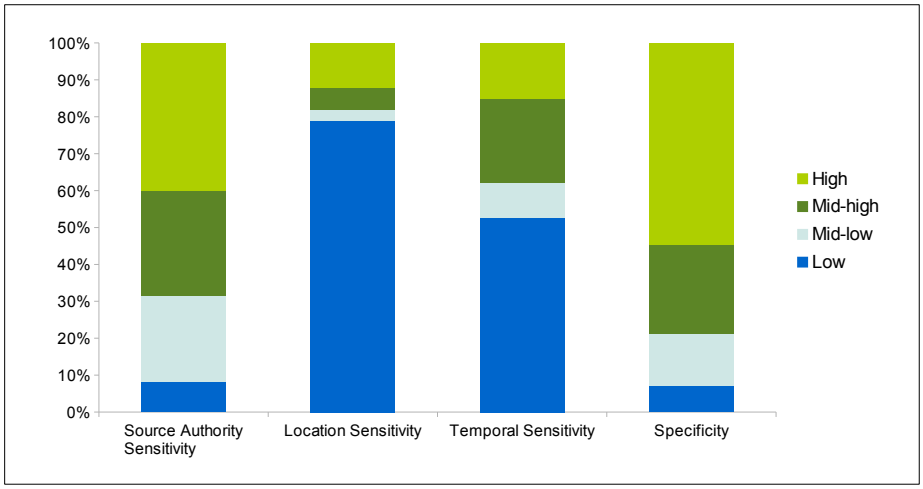


Figure 2.1: Distribution of source authority sensitivity, location sensitivity, temporal sensitivity and specificity, measured a scale from 1 (low) to 4 (high)

- There tends to be a relation between the categories *news* and *sports* on the one hand and a high temporal sensitivity on the other hand: all *news* annotated queries (8 queries) and all but one *sports* annotated queries (15 queries) were annotated with a high temporal sensitivity.
- The category *science* and the combination of the categories *health* and *science* seem to be indicators of a high source authority sensitivity. Of the 183 queries annotated with *science*, 154 were annotated with a high source authority sensitivity, and all of the 79 queries annotated with the combination *health* and *science* were annotated with a high source authority sensitivity. The category *computer* was mostly annotated with a mid-high or mid-low source authority sensitivity (215 of 250 queries).
- There seems to be a relation between the modus of the query and the location sensitivity. Of the 26 queries that were annotated with the *map* modus, 23 were annotated with a high location sensitivity.

We also found that a number of aspects of query intent were not reflected by the textual content of the query:

- There were few query words that were specifically related to the modus or the action type of the query. For example, in the queries that were annotated with the *image* modus there were no occurrences of words such as “image” or “picture”.
- Only 2 of the 90 queries that were annotated with a high temporal sensitivity contained a time-related query word.
- Of the 72 queries that included a location reference such as a city or a country, 36 were annotated with a high location sensitivity. In the remaining queries with lower location sensitivity, 11 location references occurred.

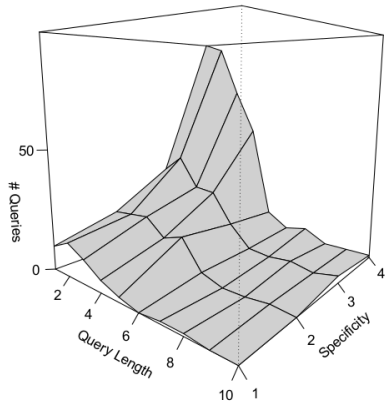


Figure 2.2: Effect of query length (number of words in query) on specificity as measured on a scale from 1 (low) to 4 (high)

Finally, we found that 54% of the queries (331 queries) were annotated with a high specificity, 56% of which consisted of only one or two words (187 queries). We found a very weak negative correlation between query length and specificity ($\tau = -0.0968$, $p = 0.0047$). Figure 2.2 shows the relation between query length and the annotated specificity of the query intent.

2.6. CONCLUSION AND FUTURE WORK

In the research described in this chapter we collected a set of queries that are labelled with their underlying intent. The queries were annotated by the searchers themselves to have a ground truth data set. This set can be used (1) to investigate the reliability of external human assessors and (2) to train automatic classification models.

The data shows that the textual content of the queries does not give many hints as to what annotation in terms of modus, action, source authority sensitivity, location sensitivity, time sensitivity and specificity can be expected. Moreover, query length does not predict the specificity of the query intent. This indicates that it might be difficult for an external human assessor that does not know the searcher or the context of the query to reliably determine what the searcher's query intent was. If it is difficult for a human assessor, it is even more difficult to assess query intent using automatic classification for a system without world knowledge or additional knowledge about the searcher.

We are interested in the differences between query intent annotation by external human assessors and the searcher's own annotations. Currently, the collected ground truth data is being labelled by external human annotators using the same annotation scheme as presented in this work. If it is possible for external annotators to reach consensus about search intent that matches the ground truth search intent, then automated classification may be possible as well. The main contribution of this

work is a resource that helps to validate the assumptions that are made by state of the art methodologies for qualitative and quantitative studies of query intent. Indeed, current state of the art studies are based on external assessors, assuming these are sufficiently close to the original intent of the searcher. Our work enables validation of this assumption.

We will use the knowledge about differences and commonalities between external assessors and the searcher for improving automatic intent classification. We expect that knowledge about the user's expertise, search history, and other computer behaviour to be the most important factors to be able to understand the intent of a query. Therefore, in future work we will address the use of the user's search history and current computer activities to assess a searcher's intents. Monitoring computer activities may provide context about the query and disambiguate its intent.

3

ASSESSING E-MAIL INTENT AND TASKS IN E-MAIL MESSAGES

Edited from: **Maya Sappelli, Suzan Verberne, Gabriella Pasi, Maaike de Boer, Wessel Kraaij** (2016) *Collecting tasks and intent of e-mail messages*, Under revision: Information Sciences.

In this chapter we propose a task-based classification of e-mail messages. The task-based classification relies on a taxonomy that we have defined and tested by assessing its reliability and validity. It consists of the following message-level dimensions: E-mail Act, Implicit Reason, Reply Expectation and Number of Tasks and the following task-level dimensions: Spatial Sensitivity, Time Sensitivity and Task Type

This taxonomy was used to annotate parts of the Enron and Avocado datasets. The annotated parts will be made available to the research community in order to stimulate research into (automated) task-based priority estimations of e-mail messages.

From the annotations we conclude that approximately half of the messages contained an explicit task. Typically only one task was conveyed per message. Furthermore, most messages are sent to deliver information or to request information. The analysis of the conversations revealed that there is a high probability that a message that was sent to deliver some information is followed by another "deliver" message. This suggests that much information is delivered, even though no request for information has been made.

The task-based annotations presented in this work can be used for research into (automatic) categorizations of messages. Detecting whether a message contains a task, whether a reply is expected, or what the spatial and time sensitivity of such a task is, can help in providing a detailed priority-estimation of the message for the recipient.

3.1. INTRODUCTION

In the project SWELL[1] we aim to develop ICT applications that minimize the risk of burn-out and improve the well-being of the employee. A large source of stress at work originates from information overload, and more specifically e-mail overload (Gantz, Boyd, and Dowling, 2009; Bawden and Robinson, 2009). Whittaker and Sidner (1996) believe that this is caused by the misuse of the original purpose of the e-mail system. The authors state that although e-mail was originally developed for the purpose of asynchronous communication, it is currently being used for task management, scheduling and personal archiving as well. This causes cluttered inboxes and information getting lost in archives.

Attempts to improve the organization of inboxes include the automatic detection of spam (Sahami et al., 1998), message categorization (Bekkerman, 2004; Chakravarthy, Venkatachalam, and Telang, 2010; Koren et al., 2011; Sappelli, Verberne, and Kraaij, 2014; Grbovic et al., 2014) and priority estimation (Aberdeen, Pacovsky, and Slater, 2010; Dredze et al., 2008; Sappelli, Verberne, and Kraaij, 2013a). Complete agents exist that help the user file messages into folders (Segal and Kephart, 1999). However, not many of these automated techniques are adopted in current systems and many users do not even use category folders at all (Koren et al., 2011; Grbovic et al., 2014). The most likely actions users make are splitting personal and work-related e-mail by using separate mailboxes (Cecchinato, Cox, and Bird, 2014) and cleaning e-mails at the end of the day (Kalman and Ravid, 2015). However, not many users spend effort on general e-mail management (deleting, moving, flagging) (Hanrahan, Pérez-Quiñones, and Martin, 2014). Nevertheless, research indicates that proper categorizations could address the problem of feeling overloaded (Whittaker and Sidner, 1996; Bawden and Robinson, 2009; Benselin and Ragsdell, 2015). The fact that categorizations are not used suggests that there may not be a full understanding of what type of categorization is needed to properly support users in the way they use e-mail.

Since e-mail clients are often used for task management (Whittaker and Sidner, 1996; Cecchinato, Cox, and Bird, 2014), we believe that task-based categorizations might be what is missing from current systems. This chapter addresses tasks in e-mail messages to better understand what the intent is behind an e-mail. In order to do so we annotate e-mail messages with both their e-mail intent and task intent. By e-mail intent we mean the intent of the sender; why did a person send the message. In that case, the intent refers to a message as a whole. Then, within a message the sender has (either implicitly or explicitly) possibly specified one or more tasks to be undertaken by the receiver. This latter aspect is referred to as the task(s) in the message. In this chapter we investigate both the intent of the message and the tasks that are conveyed in the message. Additionally, we investigate how an e-mail conversation between two individuals evolves over time. We focus on messages from person to person, as computer generated messages or newsletters for example are not likely to contain explicit tasks for the recipient.

The contribution of this chapter is three-fold. First we present a taxonomy for

[1] http://www.swell-project.net

a task based classification of e-mail messages. Second we present an annotated dataset that will be shared with the research community to provide new opportunities for the development of (automated) e-mail support systems. And third, we present an initial analysis of how senders convey tasks in e-mail messages. We answer the following research questions:

1. To what extent do corporate e-mail messages contain tasks?
2. What are the characteristics of tasks in e-mail messages?
3. How does a work-related e-mail conversation evolve?

We start this chapter with an overview of literature on the analysis of e-mail message content. Then we present the results of a pilot study where we developed our e-mail classification scheme. In this study we determine which dimensions of content analysis are reliable for annotation. Additionally, we assess the validity of using annotations by independent assessors. Next, we present the results of a larger-scale annotation study, where we annotate messages from the Enron and Avocado datasets. These datasets originate from a company setting and are likely to be representative of how tasks are conveyed in a work environment.

3.2. BACKGROUND LITERATURE

In this section we describe the literature on the analysis of e-mail message content. A limitation of e-mail research is that collections of e-mail messages are not often made publicly available. The most used publicly available dataset of e-mail messages is the Enron dataset. This is a set of messages that was made public during a legal investigation of the Enron company(Klimt and Yang, 2004). It contains over 200,000 messages. Many researchers, however, make use of their own privately collected sets of e-mail messages (Dredze, Lau, and Kushmerick, 2006; Dredze et al., 2008; Aberdeen, Pacovsky, and Slater, 2010; Kooti et al., 2015).

Some research into the content of e-mail messages has been directed at communication purposes. In an interview study, Tyler and Tang (2003) investigate the concept of the *responsiveness image* of a person in order to understand what information is conveyed by the timing of email responses. They distinguish *response expectation* (the implicit time the sender gives to the recipient to respond) from *breakdown perception* (the initiation of a follow-up action that occurs when the response expectation time has ended).

This responsiveness image could be seen as a request for attention. Hanrahan, Pérez-Quiñones, and Martin (2014) analyse responsiveness in a 2-week study by logging user interactions with e-mail and compared these interactions to diary entries of the participants. The authors propose that e-mails can be categorized into 4 groups of requests for attention: ignore, accountable non-answer (engage with message but do not reply), postponed reply and immediate reply. This categorization provides insight in both the timing as well as the type of response that is expected.

Kooti et al. (2015) add that the request of attention is not solely based on the contents of a message. They note that there is an effect of load on the replying behaviour

of people. As users receive more e-mail messages in a day, they will reply to a smaller fraction of messages.

A line of research, other than replying behaviour, that gives insight into e-mail intent is the content of the message. Gains (1999) focus on the language that is used in messages. They have analysed messages in a commercial and in an academic setting on pattern and style of the text. They found that commercial e-mail messages tend to follow standard written business English, while messages in an academic setting follow a more pseudo-conversational pattern where for example the salutation is absent.

To analyse the message style, Gains (1999) uses a classification scheme from business communication described by Ghadessy and Webster (1988). They state that there are roughly three types of business communication: informative (give information), requestive (request information) and directive (give instructions). Furthermore Ghadessy and Webster (1988) distinguish an initiate and a respond category. These categorizations seem valid descriptors for e-mail intent.

In addition to the business communication categorization, Peterson, Hohensee, and Xia (2011) assessed the formality of e-mail messages in the Enron corpus. They annotated 400 messages on a 4-point scale (very formal, somewhat formal, somewhat informal and very informal). Factors that influenced the formality of the messages were the amount of contact between sender and recipient, whether it was personal or business, the rank difference between sender and recipient, and whether the message contained a request.

In terms of the tasks in e-mail, some research stems from speech act theory. Cohen, Carvalho, and Mitchell (2004) propose to categorize e-mails according to the intent of the sender. They propose to use categories of intent based on speech act. The categories are *meeting, deliver, commit, request, amend* and *propose*. They later refine this categorization (Carvalho and Cohen, 2005), which is explained in more detail in Section 3.3.1

In addition, Lampert, Dale, and Paris (2008) have conducted several e-mail labelling experiments on Enron data to evaluate reliability of task-based intent assessments. They focus on the speech acts of request and commit. They found that the assessments were more reliable on the message level compared to the sentence level. This suggests that messages should be evaluated as a whole.

Kalia et al. (2013) not only describe the identification of tasks based on Speech Act theory, but also the tracking of tasks. They distinguish the following phases: the creation of a commitment, the discharge of a commitment, the delegation of the commitment and the cancellation of the commitment. Their algorithms require detailed NLP analysis of the message to determine what the subject, object and action is in their tasks. This is necessary to determine whether a task is delegated to another person. Their algorithms were evaluated on a selection of 4161 sentences from the Enron corpus.

In our work we will focus on both the intent-based and the task-based categorization of e-mail messages on multiple dimensions. In the next section we describe the dimensions that we take into consideration and assess the validity of those dimensions in a pilot annotation experiment with private e-mails. In Section 3.4 we

describe the annotation of two datasets of e-mail messages and assess the findings.

3.3. RELIABILITY AND VALIDITY OF E-MAIL ANNOTATIONS

From the background literature we can identify several dimensions, such as response expectation (also referred to as reply expectation), speech act and formality, on which an e-mail message can be analysed and categorized. In a pilot experiment we determine which dimensions are relevant for the assessment of *e-mail intent* and the understanding of *task conveyance* in e-mail. With *task conveyance* we mean the communication of a task for the recipient in a message. The goal is to create a reliable and valid taxonomy for a task based classification of e-mail messages. In order to do so, we assess the reliability and the validity of candidate annotation dimensions. A reliable dimension is a dimension on which two or more annotators agree in their annotations (inter-rater reliability). A valid dimension is a dimension where independent assessors typically give the same annotation as the ground truth annotation. The sender of the message determines the ground truth annotation, as his intent is the one we try to assess. The reliability and the validity of the dimensions respectively support the selection of which dimensions we should annotate in our main experiment, and whether independent assessors are actually capable of assessing the sender's intent. The reason that we assess this in a pilot experiment where the messages are not publicly available, is that we need to involve the original senders of the messages to assess the validity of the proposed annotation scheme. We do not have this possibility for the datasets that we use in the main experiment. This is a limitation, since the pilot study only includes a limited number of e-mail messages because of the labour-intensive nature of the annotation work.

3.3.1. E-MAIL CLASSIFICATION SCHEME

In the selection of the dimensions for our e-mail classification scheme we have focused on those dimensions that are related to the content of the message, and more specifically the tasks that are conveyed through sending the message. On the one hand these dimensions are related to the message as a whole; what was the intent of the sender, what implicit reason was there for sending the message etc. On the other hand the dimensions are related to explicit tasks that are mentioned in the message; what is the recipient supposed to do after reading the message, how many tasks are mentioned, what is their spatial and time sensitivity and what kind of task is it.

A similar type of research has been done by Verberne et al. (2013) and we will use the same approach. They developed a detailed scheme to assess the intent behind a query entered in a web search engine. Many of the dimensions they assess seem relevant for e-mail intent and task classification as well. More specifically, the action category which is based on the taxonomy by Broder (2002): informational, transactional and navigational, bares a strong resemblance to the categories of business communication: informative, requestive and directive (Ghadessy and Webster, 1988), and to the e-mail acts defined by Carvalho and Cohen (2005).

For that reason, we evaluate *e-mail act* as one of the dimensions in our classification scheme. Other dimensions from literature that we will evaluate are *response ex-*

pectation (Hanrahan, Pérez-Quiñones, and Martin, 2014) and *source authority* (Verberne et al., 2013). These are related to the message as a whole. On the message level we also evaluate the new dimensions *implicit reason* and *number of tasks*. The detailed description of the dimensions can be found in Table 3.1.

For each task that is conveyed in a message we evaluate the following dimensions based on the query intent literature (Verberne et al., 2013): *spatial sensitivity*, *time sensitivity*, *task specificity* and *task topic*. Additionally we evaluate the new dimensions *task type* and *task subject*. A detailed description of these dimensions can be found in Table 3.2.

Dimension	Description
E-mail Acts(Carvalho and Cohen, 2005)	What are the two main e-mail acts in the message? This dimension has categorical values [2], consisting of: *request*: A request asks (or orders) the recipient to perform some activity. A question is also considered a request (for delivery of information) *propose*: A propose message proposes a joint activity, i.e., asks the recipient to perform some activity and commits the sender as well, provided the recipient agrees to the request. A typical example is an email suggesting a joint meeting *commit*: A commit message commits the sender to some future course of action, or confirms the sender's intent to comply with some previously described course of action *deliver*: A deliver message delivers something, e.g., some information, a PowerPoint presentation, the URL of a website, the answer to a question, a message sent "FYI", or an opinion *amend*: An amend message amends an earlier proposal. Like a proposal, the message involves both a commitment and a request. However, while a proposal is associated with a new task, an amendment is a suggested modification of an already-proposed task *refuse*: A refuse message rejects a meeting/action/task or declines an invitation/ proposal *greet*: A greet message thank someone, congratulate, apologize, greet, or welcomes the recipient(s) *remind*: A reminder message reminds recipients of coming deadline(s) or threats to keep commitment

[2] descriptions taken from Carvalho and Cohen (2005)

Response Expectation (Hanrahan, Pérez-Quiñones, and Martin, 2014).	What type of response is expected? This dimension has ordinal values, consisting of: *ignore*: There is no realistic expectation that the recipients will properly read the email, let alone respond to them *accountable non-answer*: Recipient is expected to engage with the message or its attachments, but there is no reply required *postponed reply*: The messages requires a reply but not immediately *immediate reply*:] The message requires a reply as soon as possible
Source authority (Verberne et al., 2013)	4-point ordinal scale (very low, low, high, very high): What is the authority of the sender?
Implicit Reason	What was the reason to send the message? This categorisation is based on the task-related categories in Enron[3], consisting of: *administrative procedure*: The message is part of an administrative procedure, such as financial arrangements or the organization of a meeting *legal procedure*: The message is part of a legal procedure *internal collaboration*: The message is part of a collaboration between people within the same company, such as messages related to internal projects *external collaboration*: The message is part of a collaboration between people that are not working for the same company *travel planning*: The message is part of a travel plan, such as a confirmation of a hotel booking *employment arrangements*: The message is about employment arrangements, such as messages related to job seeking or job applications *logistic arrangements*: The message is about logistic arrangement. This includes general support and technical support *personal*: The message is of a personal, non work related, nature *other*

[3] retrieved from http://bailando.sims.berkeley.edu/enron_email.html

Number of tasks	How many tasks for the recipient are explicitly stated in the message (typically a number between 0 and 10)?

Table 3.1: Dimensions related to the intent of e-mail messages; what was the motivation of the sender to send the message

Dimension	Description
Spatial sensitivity (Verberne et al., 2013)	4-point ordinal scale (very low, low, high, very high): Is the task associated with a certain location? For example is a meeting supposed to take place at a certain location, then the spatial sensitivity is very high; the task can only be executed there.
Time sensitivity (Verberne et al., 2013)	4-point ordinal scale (very low, low, high, very high): Is the task associated with a certain time? For example is the task supposed to be executed at a certain time, then the time sensitivity is very high.
Task specificity (Verberne et al., 2013)	4-point ordinal scale (very generic, somewhat generic, somewhat detailed, very detailed): How detailed is the description of the task?
Task type	What is the type of the task? categorical, consisting of: *physical*: The task requires physical action. For example 'Do the groceries' or 'Get flowers' *informational*: The task requires knowledge. For example 'When was Einstein born?' or 'Can you write a report about Einstein?' *procedural*: The task has a procedural nature; it is mainly administrative. For example 'Can you plan a meeting'
Task subject	What is the subject of the task/ What is the task about? categorical, consisting of: *product*: e.g. 'Get flowers' *service*: e.g. 'Fix this problem for me *acknowledgement*: e.g. 'Write me a recommendation letter' *announcement*: e.g. 'Send a message that the meeting location has changed *decision*: e.g. 'Decide which flowers you prefer?' *reservation*: e.g. 'Confirm my reservation for room X' *event*: e.g. 'Make a schedule for event X' *meeting*: e.g. 'Confirm that you can meet at 10.30' *instructions*: e.g. 'Provide instructions how I can solve this bug' *collaboration*: e.g. 'Ask company X if they want to collaborate on topic Y' *information*: e.g. 'Provide the birth date of Einstein'

	other
Task Topic (Verberne et al., 2013)	categorical, fixed set of categories from the well-known Open Directory Project (ODP), giving a general idea of what the topic of the task is.

Table 3.2: Dimensions that describe a task that is to be undertaken by the recipient of the message and which was specified explicitly by the sender

3.3.2. METHOD

In order to answer our research questions about reliability and validity of message intent assessments we calculate the agreement between assessors. In our research we distinguish three types of assessors based on their relation to the e-mail message: 1) the assessor was the sender of the message, 2) the assessor was the recipient of the message, or 3) the assessor has no relation to the message (independent).

For this experiment, 5 collaborators have provided a total of 50 e-mail messages from their correspondence with the other collaborators. Each of them filled out a spreadsheet with columns corresponding to the various dimensions. The rows of the spreadsheet corresponded to the messages for which he or she was either the sender or the recipient. Furthermore, one independent individual (non-collaborator) who was not familiar with the context of the messages was asked to fill in the spreadsheet as well. All individuals were given the instructions for the dimensions as presented in Table 3.1 and Table 3.2.

In this experiment we focus on two aspects of the annotations; *reliability* and *validity*, which we describe further in the next subsections.

RELIABILITY

First we assess the agreement between assessors to determine the reliability of each dimension. Here we do not look at the relation of the assessor to the message (sender, recipient or independent). We calculate the inter-annotator reliability; how often do two annotators agree on their annotations for a dimension. The agreement on the dimensions was calculated using Cohen's kappa (Cohen, Carvalho, and Mitchell, 2004). For the ordinal dimensions the agreement was calculated using weighted Kappa (Cohen, Carvalho, and Mitchell, 2004). The ordinal dimensions are: response expectation, source authority, number of tasks, spatial sensitivity, time sensitivity and specificity. All kappa-agreements in this chapter are interpreted using the scale by Landis and Koch (1977), where a κ between 0.01–0.20 can be seen as slight, 0.21–0.40 as fair, 0.41–0.60 as moderate, 0.61–0.80 as substantial and >0.80 as almost perfect agreement.

VALIDITY

Secondly, we assess the difference in agreement between sender–recipient assessor pairs and sender–independent assessor pairs to assess the validity of the dimension. Here we take the role of the assessor into account; was he the sender of the message, the recipient of the message or did he have no relation at all with the message. With this research we assess whether independent assessors can correctly interpret the

original intent of the message. In this latter case we assume that the sender of the message knows his intent, so his annotation is the ground truth.

To assess the validity of the assessment between sender–recipient annotator pairs and sender–independent pairs we use the pair-wise nature of the data (each message has been assessed by two pairs of annotators). We cannot use Cohen's κ because it aggregates annotations over a complete dataset and cannot measure the agreement between two annotations for a single message. Therefore, we take the approach by Verberne et al. (2013) where we compute per message a vector of scores for each of the assessor type pairs. For a given message, the annotation similarity between the two assessors of an annotator pair consists of Jaccard scores for categorical dimensions and normalized distances for the ordinal dimensions. Then we perform a pairwise significance test to compute the difference between annotation similarity by sender–recipient pairs and sender–independent pairs.

3.3.3. RESULTS

We begin with an overview of the distribution of annotations. In this dataset, most messages contained a single task (55.6%), while 35.6% contained no task at all. There were no messages with more than 3 tasks. The main e-mail act was to deliver information (52.2%), followed by a request (21.7%). There was not often a necessity for immediate reply (6.5%): 37% required a postponed reply, while 56.6% required an accountable non-answer. The implicit reason for sending the message was mostly collaboration: 34.1% external collaboration and 43.2% internal collaboration.

More than half of the tasks were informational in nature (63.3%), while the remaining tasks were often procedural (30%). This is confirmed by the subject of the tasks that was often information (53.6%) or a decision (14.3%). Other common subjects of tasks were meeting, product or service (7.1% each).

RELIABILITY

Table 3.3: Agreement on the e-mail intent dimensions for sender–recipient (SR) pairs and sender–independent (SI) pairs. * indicates significance of the kappa value at the 0.05 level

Dimension	κ SR	κ SI
1st E-mail Act	0.230*	0.346*
2nd E-mail Act	0.285*	0.147
Response Expectation	0.649*	0.574*
Source Authority	0.263*	0.160
Implicit reason	0.021*	0.000
Number of tasks	0.664*	0.556*

To answer the question which dimensions can be assessed reliably we look at the inter-annotator agreement. Dimensions where each annotator pair has at least a fair agreement are considered as reliable. In Table 3.3 we present the agreement between sender–recipient (SR) and sender–independent (SI) on the dimensions related to e-mail intent. We see a fair agreement on the first e-mail act, for both sender–recipient and sender–independent pairs. The agreement between sender and independent

assessor on the second e-mail act was not significant, because there were too few annotations of the second e-mail act made by the independent assessor.

The agreement on response expectation is substantial for sender–recipient and moderate for sender–independent. This suggests that although an independent assessor can reliably estimate the response expectation, it is even easier for the recipient of a message.

In terms of source authority we see a fair agreement between sender and recipient, while the agreement between sender and independent assessor is slight and not significant. Since the agreement is low for the sender–independent pair (0.160) we decided to exclude this dimension from further experiments.

The implicit reasons in the message were assessed with only slight agreement between sender and recipient. The agreement between sender and independent assessor could not be calculated reliably as there was not enough variation in the annotations of the independent assessor compared to the annotations of the sender for the amount of data. On the basis of these agreements we should also remove the implicit reason dimension from further experiments. A detailed analysis reveals that the main reason for the low agreement is because of disagreement whether a message is considered to be external or internal collaboration. The distinction between the categories can be made by looking at the employer of the sender and comparing it to the employer of the recipient. Often this information can be extracted from the e-mail addresses. In this experiment, however, this information was not available, which made it difficult for the independent assessor to assess this dimension. We decided to keep the dimension in further experiments, but make the e-mail addresses of sender and recipient part of the data.

The agreement on the number of tasks in the message is substantial between sender–recipient and moderate for sender–independent.

Table 3.4: Agreement on the task dimensions for sender–recipient (SR) pairs and sender–independent (SI) pairs. * indicates significance of the kappa value at the 0.05 level. This data was evaluated on 28 tasks

Dimension	κ SR	κ SI
Spatial Sensitivity	0.362*	0.421*
Time Sensitivity	0.658*	0.325*
Specificity	0.230*	-0.211
Type	0.563*	0.356*
Subject	0.221*	0.000
Topic	-0.032	-0.004

In Table 3.4 we present the agreement between sender–recipient and sender–independent on the dimensions related to the tasks in the e-mail messages.

There was a fair to moderate agreement on the spatial dimension. On the time sensitivity of tasks, the agreement between sender and recipient was much higher (substantial) than between sender and independent assessor (fair). This suggests that it is difficult for an independent assessor to reliably estimate the time sensitivity of a task. An explanation can be that the time assessment is made based on implicit information such as the past expectations between sender and recipient.

The agreement on the specificity of the task was fair between sender and recipient, but negative between sender and independent assessor. Comments revealed that the assessors could not come to consensus about the interpretation of specificity, making this dimension hard to assess. Therefore this dimension was excluded in the remaining experiments.

The agreement on the type of the task was moderate between sender and recipient and fair between sender and independent assessor.

The agreement on the subject of the task could not be calculated between sender and independent assessor as there were too little data points for the number of categories in the dimension. Between sender and recipient the agreement was fair. On the basis of these results we have excluded the task subject dimension.

The agreement on the topic of the task was very low and not significant for both pairs of assessors. Since there was little variation in the general topic categories that could be assigned this dimension was excluded in the remaining experiments.

VALIDITY

Table 3.5: Difference in agreement between sender-recipient (SR) pairs and sender-independent (SI) pairs on message dimensions. Reported Jaccard scores are averaged over all messages

Dimension	Jaccard SR	Jaccard SI	p-value SR-SI	Cohen's d
1st E-mail Act	0.52	0.63	0.23	0.22
2nd E-mail Act	0.61	0.20	0.00	0.93
Response Expectation	0.76	0.67	0.36	0.19
Implicit reason	0.44	0.65	0.02	0.45
Number of tasks	0.92	0.91	0.64	0.09

To answer the question whether an independent assessor can assess the intent of a sender just as well as the recipient of a message, we assessed the difference in agreement between sender–recipient (SR) pairs and sender–independent (SI) pairs. This was calculated in a pair-wise fashion on message level as described in Section 3.3.2.2. The differences in agreement scores, significance values and effect size in terms of Cohen's d are reported in Table 3.5. From this we can conclude that an independent assessor is capable of interpreting the intent of the sender just as good as the recipient for the dimensions 1st E-mail act, response expectation, source authority and number of tasks ($P > 0.05$, so no significant difference between SI and SR).

However, the independent assessor is not capable of interpreting the implicit reason as good as the recipient can. Therefore we should be careful in drawing conclusions based on this dimension.

The differences in agreement scores, significance values and effect size in terms of Cohen's d for the task dimensions are reported in Table 3.6. From this we can conclude that an independent assessor is capable of interpreting the tasks that the sender conveyed just as good as the recipient for all task dimensions. Nevertheless, caution should be taken when depending on the time dimension as the significance and effect size values indicate that it might be a dimension that is difficult to assess by an independent assessor.

Table 3.6: Difference in agreement between sender-recipient (SR) pairs and sender-independent (SI) pairs on task dimensions. Reported Jaccard scores are averaged over all messages

Dimension	Jaccard SR	Jaccard SI	p-value SR-SI	Cohen's d
Spatial Sensitivity	0.83	0.83	1.00	0.00
Time Sensitivity	0.88	0.77	0.09	0.61
Type	0.77	0.73	0.72	0.05

3.4. E-MAIL INTENT ASSESSMENTS ON LARGER DATASETS

In this second experiment we annotated part of a public dataset, Enron, and part of a licensed dataset, Avocado, of e-mail messages. We used the same classification scheme as described in Section 3.3.1 excluding the categories *Source Authority*, *Task Specificity*, *Task Subject* and *Task Topic*. These were excluded based on the results of the experiment described in Section 3.3.

Details on the datasets can be found in Section 3.4.1. The aim of this experiment is to analyse task conveyance in e-mail message. Furthermore the annotated dataset will be shared with the research community to provide new opportunities for the development of (automated) e-mail support systems.

3.4.1. DATA COLLECTION

The data that we have annotated have been selected from the Enron and Avocado collections. The Enron dataset is a set of messages that was made public during a legal investigation of the Enron company (Klimt and Yang, 2004). It contains over 200,000 messages from 158 users that were sent or received between 1998 and 2004. The Enron company was an American energy, commodities and services company.

The Avocado collection[4] is a set of over 800,000 e-mail messages from the mailboxes of 279 users that were sent or received between 1995 and 2003. The data is collected from a defunct information technology company referred to as "Avocado".

Enron We selected a total of 1145 messages from the Enron dataset. Of these messages, 750 were randomly selected from the sent messages of the 15 most active users (50 each). The remaining 395 were coming from 15 randomly selected complete conversations. A conversation consists of all the messages sent between two individuals. These can contain multiple threads. Ten of the conversations were between two individuals within Enron, while 5 conversations were between an Enron-employee and an outsider. The average length of the selected internal conversations was 33.1 messages (minimum 6, maximum 81 messages). The external conversations had an average length of 14.4 (minimum 3, maximum 41 messages).

Each selected message was annotated according to the scheme in Section 3.3.1 using Amazon Mechanical Turk. Each message was annotated by 2 workers in order to make it possible to assess the agreement. The annotators were required to have an annotation acceptance rate of more than 95% to ensure quality. A total of 3 messages were excluded from the final dataset because of noisy annotations, resulting in

[4]https://catalog.ldc.upenn.edu/LDC2015T03

a dataset of 1143 annotated e-mail messages.

Avocado A total of 379 messages was selected from the Avocado dataset. Of these messages, 250 were randomly selected from the sent messages of the 5 most active users (50 each) of which 7 messages were excluded because they were duplicates. The remaining 136 messages originated from 5 randomly selected complete conversations. A conversation consists of all the messages sent between two individuals. These can contain multiple threads. Of these conversations, 3 were between employees of Avocado, and 2 were between an Avocado-employee and an outsider. The average length of the selected internal conversations was 35 messages (minimum 10, maximum 71 messages). The external conversations had an average length of 16 (minimum 13, maximum 19 messages).

Each selected message was annotated according to the scheme in Section 3.3.1. Because of license agreement, this set could not be annotated using Amazon Mechanical Turk. Instead, the data was annotated by two expert annotators, who discussed the annotation dimensions in detail prior to annotating. A subset of 204 messages was annotated by both annotators to assess agreement. The remaining items were only annotated by one expert annotator

3.4.2. RESULTS

In this section we describe the analysis of the results of the annotations. We focus on assessing the agreement, present frequency distributions and transition graphs for the dimensions of interest.

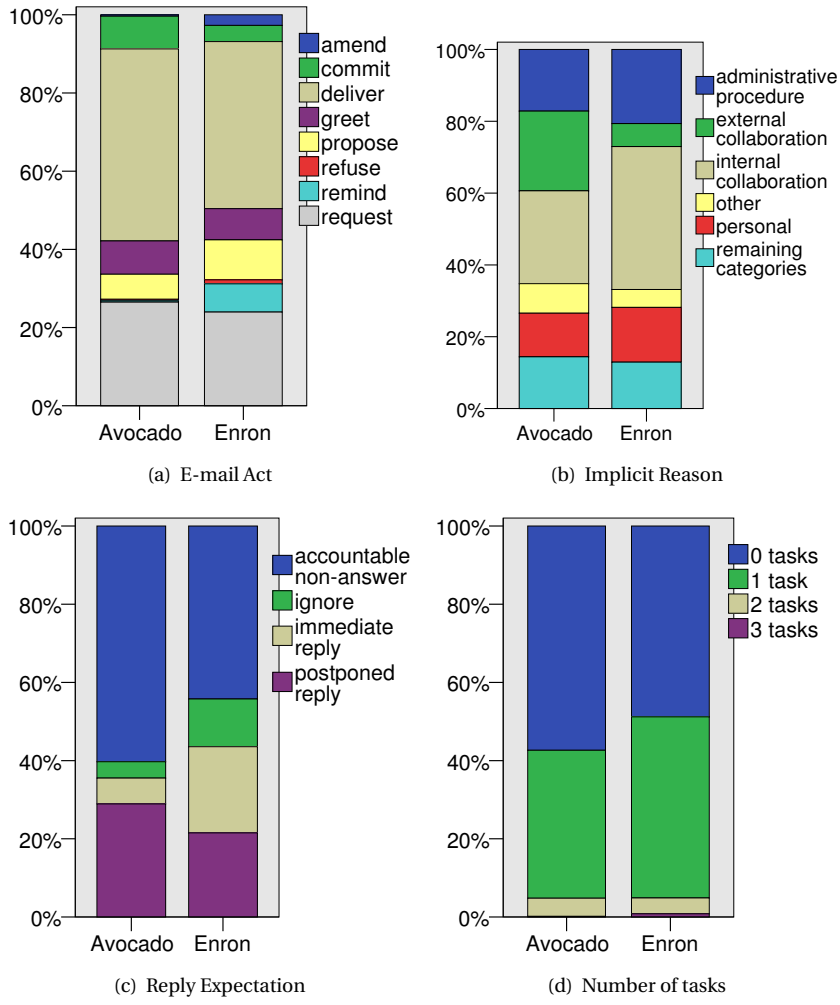
E-MAIL INTENT

We analyse the annotations on the message level. These dimensions are related to the e-mail message as a whole.

Table 3.7: Inter-annotator agreement on the e-mail intent dimensions. All are significant at the 0.05 level

Dimension	κ Enron	κ Avocado
1st E-mail Act	0.319	0.585
Reply Expectation	0.328	0.610
Implicit reason	0.228	0.217
Number of tasks	0.334	0.727

When we look at the agreement on the mail dimensions, the results show that for the Enron set the agreements are all fair. For the Avocado set the agreement is higher, being moderate or even substantial for all dimensions except the Implicit Reason, which has fair agreement. An analysis of the annotations shows that this dimension has a lower agreement because of confusion between the *logistic arrangements* category and the *internal collaboration* category. The main activities in the Avocado company seem to be of a supportive and programmatic nature. Therefore many messages can actually be seen as both logistic as well as collaboration. We have not reported the agreement on the 2nd e-mail act dimensions, as there were insufficient assessments made on the Enron dataset to assess it properly.



(a) E-mail Act

(b) Implicit Reason

(c) Reply Expectation

(d) Number of tasks

Figure 3.1: Distribution of mail dimensions in the Enron dataset compared to the Avocado dataset

The high agreement on the Avocado set suggests that expert annotators reach higher agreement than non-expert annotators. Another explanation for the high agreement is that the messages in the Avocado set are easier to categorize. For both datasets the agreement is high enough to establish that the categorization can be assessed with at least a fair reliability. We cannot assess the validity of the assessments as the original senders and recipients are not available as assessors.

Figure 3.1 shows the distribution of the annotation of the e-mail dimensions for the Enron and Avocado datasets. We see that the distributions of the e-mail acts and reason for sending the message are very similar between the datasets. The e-mail acts *amend* and *refuse* are hardly ever used as main act in the message. There are

also a few implicit reason categories that are not used very often: *Travel Planning* and *Other*. In the Avocado set, *Legal Procedures* do not occur, where the Enron set contains a couple of messages related to legal procedures. This can be explained by legal issues that were surrounding Enron specifically.

The reply expectation reveals that Enron messages are often a bit more urgent than Avocado messages (22% immediate reply vs. 7% immediate reply). Avocado messages are read without a reply in 61% of the cases whereas this is only 44% in Enron. When we look at task conveyance, the Avocado messages contain explicit tasks less often than the Enron messages (43% vs. 51%). Overall half of the messages do not contain a task. If the message does contain a task, it contains typically no more than one task.

For Enron we have information available about the various roles of the senders. We selected data from 2 directors, 5 employees, 1 manager, 1 trader and 4 vice presidents. These results are presented in Figure 3.2. Here we can see that there are only slight variations in response expectation and number of explicit tasks based on employee role. We do see that normal employees seem to have a higher number of internal collaboration based messages, whereas managers send more employment arrangement related messages. Directors and vice presidents engage in more administrative procedures than normal employees and traders. Note, however, that as the data of the manager is only from 1 individual, these results are not generalizable. Overall the findings are not surprising and seem to comply with intuitions about office work.

TASK INTENT

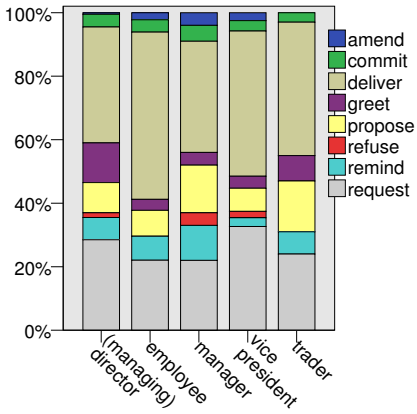
In this section we start again with the assessment of the inter-annotator agreement. We follow with an analysis of the distributions of annotations.

Table 3.8: Inter-Annotator Agreement on the task intent dimensions. These results are for the most prominent task in a message as there are few messages with more than 1 task. All are significant at the 0.05 level

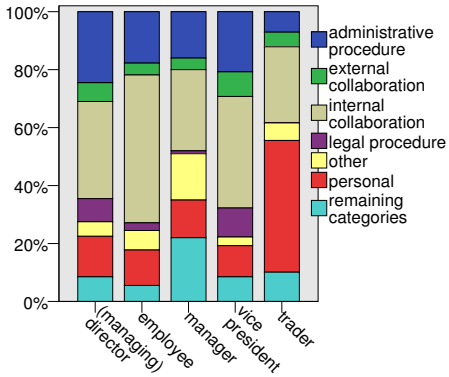
Dimension	κ Enron	κ Avocado
Spatial Sensitivity	0.187	0.451
Time Sensitivity	0.138	0.534
Type	0.180	0.355

For the task dimensions we see slight to fair agreements on the Enron set, and fair to moderate agreements on the Avocado set (Table 3.8). This suggests that the task dimensions are harder to assess reliably, but that this can be improved with training (discussing the dimensions before annotation, like the expert annotators did). These agreements are calculated over the number of messages that contain at least one task, which are 379 messages for Enron and 79 messages for Avocado.

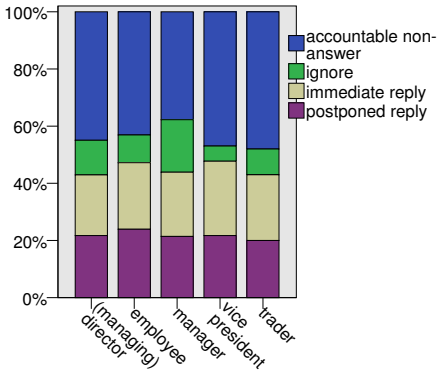
The distribution of the annotations of the task dimensions are presented in Figure 3.3. It shows that the tasks in the Avocado set are typically less spatial and time sensitive than the tasks in Enron messages. The Enron messages contain physical tasks more often, whereas the percentage of procedural tasks in the Enron and Avocado messages are almost equal. Furthermore, the Avocado company seems to be



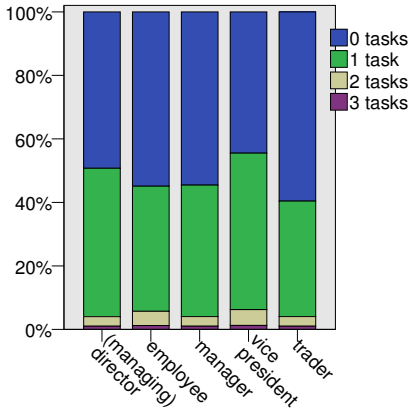
(a) E-mail Act



(b) Implicit Reason

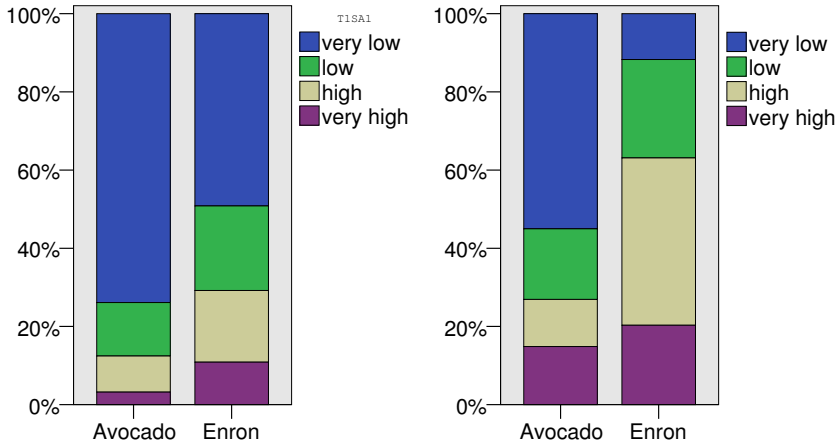


(c) Reply Expectation



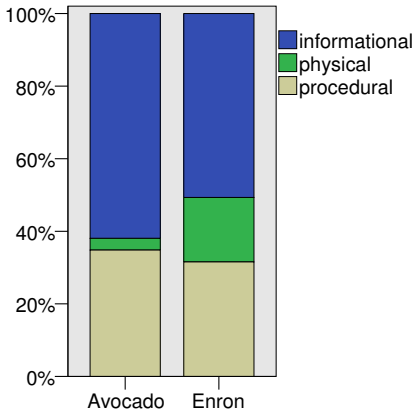
(d) Number of tasks

Figure 3.2: Distribution of mail dimensions per role in the Enron dataset



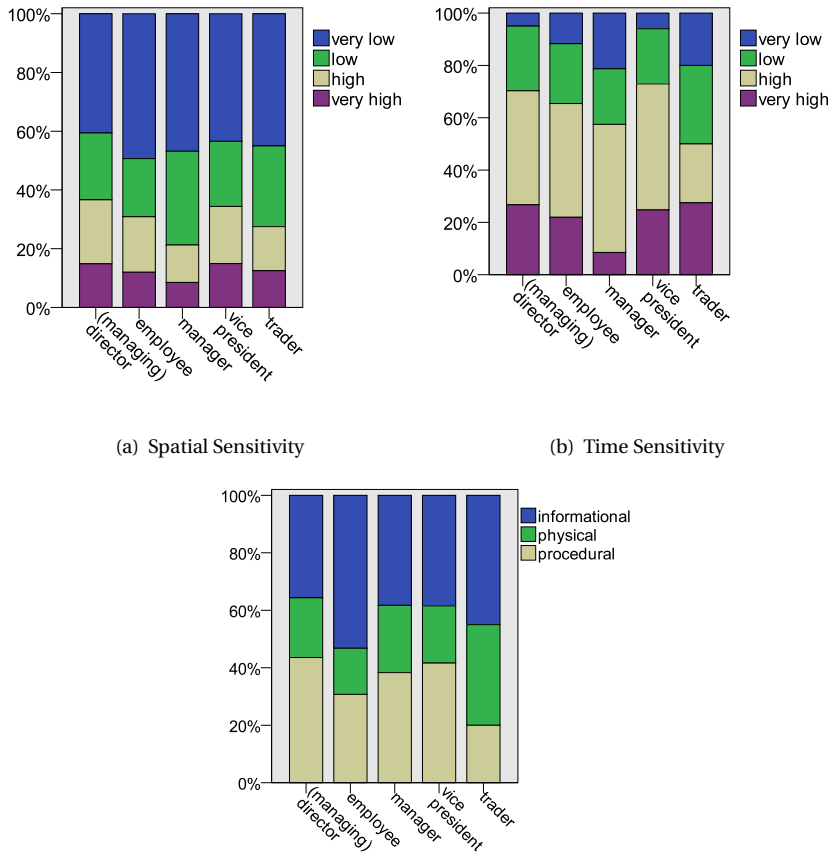
(a) Spatial Sensitivity

(b) Time Sensitivity



(c) Task Type

Figure 3.3: Distribution of task dimensions in the Enron dataset compared to the Avocado dataset



(a) Spatial Sensitivity

(b) Time Sensitivity

(c) Task Type

Figure 3.4: Distribution of task dimensions per role in the Enron dataset

a bit more service oriented in their tasks, whereas Enron employees seem to be in meetings more often.

When we look at the task dimensions per role in the Enron dataset (Figure 3.4) we see that employees send mostly informational tasks via e-mail. Vice presidents and managers send equally many procedural and informational tasks.

EVOLUTION OF A CONVERSATION

In this section we look at how conversations evolve. We do this by analysing e-mail acts and count how often they follow each other in a conversation. An example of part of a conversation can be found in Table 3.9. It is important to realize that a conversation includes all e-mail messages between two individuals. This is different from the so-called threads that are used in e-mail clients. A thread consists of all the messages between two individuals where the subject is the same, or with a prefix

Table 3.9: Part of a conversation from Enron

Sender	Recipient	Date	Subject	Body
Stan	Jim	9-11-2001 13:55	RE: Notre Dame	I would like you to get directly involved in selling the assets we have targeted for next year while overseeing the operations of Mariella, Pete and Orlando. That is a lot of stuff. However I will keep you in mind as we figure out who is going to be on the transition team.
Jim	Stan	19-11-2001 10:29	FW: Draft MOU regarding equity sale	Wade/Rob/Bruce:I think we should reconfigure the MOU to constitute a binding obligation to purchase and sell rather than a MOU that would lead to a definitive agreement. We are trying to force an answer, it seems to me at this stage of the game, a definitive offer should be put forward for acceptance or rejection.Thoughts? Jim
Jim	Stan	19-11-2001 15:06	FW: ASSET SALES MEETING WITH STAN HORTON	Stan:Shouldn't Ray Bowen or Jeff McMahon participate? Jim
Stan	Jim	20-11-2001 05:28	RE: ASSET SALES MEETING WITH STAN HORTON	I thought we should review the process with Mark and Jeff first. I will go ahead and invite either Jeff or Ray.

such as RE: or FWD:. A conversation can span multiple threads. In Figure 3.5 we present the transition in e-mail acts for the conversations in the Enron and Avocado datasets.

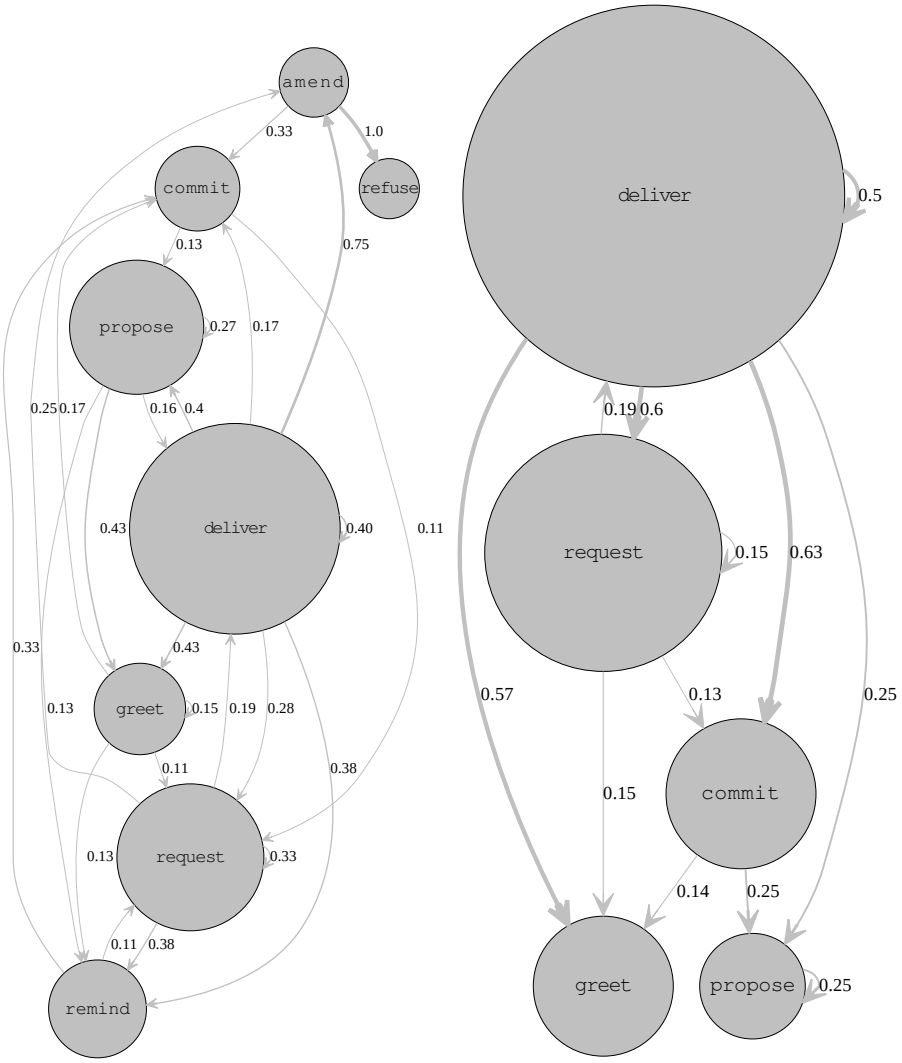
For Enron it is interesting to note that a *request* message is often followed by a *deliver* message, a *commit* message or a *remind* message (Figures 3.5(a)). This seems logical as the recipient can either full fill the request, commit to doing so at another time. When that does not happen the sender can remind him about the request. Furthermore it is interesting to see that a *deliver* message or a *propose* message are often followed by a *greet*-type message. This could for example be a thank-you message. Another noteworthy point is that a refusal of a request seems to only occur after a request has been amended.

Similar trends seem to be going on in the Avocado messages: *requests* are mostly delivered or being committed to, and *deliver* messages are often followed by *greet* messages (most likely a thank you message). In contrast to Enron *greet* messages also follow after *request* and *commit* messages. Some e-mail act annotations, such as *refuse*, did not occur in the selection from the Avocado dataset.

Another interesting aspect to note in both Enron and Avocado is that there is a high probability that a *deliver* message is followed after another *deliver* message. This suggests that much information is delivered, even without a request. This can be for example because of own initiative, an earlier commitment, or because of agreements made outside the e-mail communication.

3.5. DISCUSSION AND LIMITATIONS

In this chapter we described the annotation of two e-mail datasets for the research community in terms of message intent and task conveyance. We started with a pilot



(a) Enron (b) Avocado

Figure 3.5: Transitions in E-mail Acts in the Enron dataset compared to the Avocado dataset. Size of the ellipse represents the frequency of occurrence, the numbers represents the transition probabilities

study in which we assessed the reliability and the validity of the various dimensions that we took under consideration to describe e-mail messages. This pilot study has resulted in a taxonomy for task-based e-mail classification that consists of the following dimensions on message level (e-mail intent): *E-mail Act*, *Implicit Reason*, *Reply Expectation* and *Number of Tasks*. It consists of the following dimensions on the task level: *Spatial Sensitivity*, *Time Sensitivity* and *Task Type*.

The limitation of the reliability and validity research as we have presented it in the pilot study, is that it was assessed on only a small dataset of 50 e-mail messages. The reason is that e-mail messages for research are hard to obtain because of the privacy concerns involved. To assess the reliability and validity we also need access to both the sender, the recipient and independent assessors, which makes it even harder to find useful messages. Finally, the task of annotating e-mail messages is labour intensive.

Despite the small dataset in the pilot study, we were able to obtain significant results. From these we can conclude that it is possible to assess the intent of a message and the tasks that were conveyed by independent assessors on all the dimensions in the taxonomy except for *Implicit Reason*. This suggests that the assessment of e-mail intent and tasks in e-mail messages is easier than query intent as there were few valid dimensions found by Verberne et al. (2013) for query intent. This is not surprising, as an e-mail message contains much more textual content than a query to base the assessment on.

In the main experiment, we annotated selections of the Enron and Avocado datasets using the task-based taxonomy. The limitation here is that we could not assess the validity as the original senders of the messages were not available. Furthermore, because of the strict license agreements for the Avocado dataset we could not use crowd-sourcing for annotation. This forced us to only annotate a very small portion of the messages (less than 1% of the messages). Nevertheless, the similarities between the distributions of the two datasets do not give reasons to doubt the representativeness of the data.

A final limitation is in the analysis of the conversations. The conversations that were selected may not have been complete. Senders and recipients may have deleted messages before they were collected in the Enron or Avocado datasets. This may have distorted the analysis of the conversations.

These limitations show the challenges in working with e-mail data. E-mail messages are very sensitive to privacy concerns. Moreover the datasets are often incomplete because of messages that are deleted. Still, the annotations are an important contribution to the field. More effort should be taken to develop good annotated datasets of e-mail messages that can be shared with the research community, in order to make it possible to compare results.

3.6. CONCLUSION AND FUTURE WORK

In this chapter we presented a new taxonomy for an intent-based and task-based classification of e-mail messages. This taxonomy consists of the dimensions *E-mail Act*, *Reply Expectation* and *Number of Tasks* that are assessed at the message level. The dimension *Implicit Reason* is assessed at the message level as well, but its reli-

ability and validity should be investigated further. The task dimensions in the taxonomy consist of the dimensions *Spatial Sensitivity*, *Time Sensitivity* and *Task Type*. These are assessed for each task that is identified in an e-mail message.

The taxonomy was used in an annotation experiment with a selection of messages from the Enron and Avocado e-mail datasets. The resulting annotated datasets are available for future research[5].

Finally, we presented a number of analyses of the annotated data. These allow us to answer our research questions. The first research question was "To what extent do corporate e-mail messages contain tasks"? We can conclude that approximately half of the e-mail messages contain a task. Moreover, typically only one task at a time is conveyed in a message. Furthermore, most messages are sent to deliver information or to request information. Requests are not often rejected. The implicit reason for sending messages is typically because of general collaboration, an administrative procedure or personal reasons. In terms of reply expectation, about half of the messages do not require a reply. If a reply is needed, it typically does not need to be immediate.

For our second research question "What are the characteristics of tasks in e-mail messages?" we can conclude that most tasks can be executed everywhere (low spatial sensitivity). Some tasks do have a high or very high time sensitivity such as a deadline, but the likeliness of this happening depends strongly on the company. The type of the task is mostly informational or procedural. This is not surprising, as both e-mail datasets are collected in a knowledge worker environment where the exchange of information is an important part of the work.

Finally, about the third research question "How does a work-related e-mail conversation evolve?" we can conclude that there is a high probability that a *deliver* message is followed after another *deliver* message. This suggests that much information is delivered, even without a request. Furthermore, *requests* are mostly delivered or being committed to, and *deliver* messages are often followed by *greet* messages. These *greet* messages are most likely thank you messages.

The annotations on these datasets can be used for research into (automatic) task-based categorizations of messages. Detecting whether a message contains a task, whether a reply is expected, or what the spatial and time sensitivity of such a task is, can help in providing a more detailed priority-estimation of the message for the recipient compared to existing work (Aberdeen, Pacovsky, and Slater, 2010; Sappelli, Verberne, and Kraaij, 2013a). Such a priority-based categorization can support knowledge workers in their battle against e-mail overload. For this reason, future work should be directed at the automatic classification on the dimensions in the taxonomy. This requires research to which dimensions can be assessed using machine learning techniques, which features optimally model each dimension and which classifiers are best suited for the task.

[5] http://cs.ru.nl/m̃sappelli/data/

4

COLLECTING A DATASET OF INFORMATION BEHAVIOUR IN CONTEXT

Edited from: **Maya Sappelli, Suzan Verberne, Saskia Koldijk, Wessel Kraaij** (2014) *Collecting a dataset of information behaviour in context.* In: Proceedings of the 4th Workshop on Context-awareness in Retrieval and Recommendation (CARR @ ECIR 2014).

We collected human–computer interaction data (keystrokes, active applications, typed text, etc.) from knowledge workers in the context of writing reports and preparing presentations. This has resulted in an interesting dataset that can be used for different types of information retrieval and information seeking research. The details of the dataset are presented in this chapter.

4.1. INTRODUCTION

This research project is part of the project SWELL[1] (Smart Reasoning Systems for Well-being at home and at work). Our overall objective is to increase the physical and mental well-being of knowledge workers[2]. We monitor their behaviour and provide them with an unobtrusive digital assistant that advises them about fitness-improving and stress-reducing behaviour.

In light of this project, we have collected a dataset of human–computer interactions during typical knowledge worker's tasks (Koldijk et al., 2013). In this data we find a large body of natural search behaviour data. Together with the detailed information of the user's computer activities, we think that this dataset is interesting

[1] http://www.swell-project.net

[2] A knowledge worker is a person whose job involves handling or using information. Nowadays, almost all office jobs are knowledge worker jobs.

"The aim of the experiment is to observe a knowledge worker during his/her tasks. To set a realistic scene you will be working on some typical knowledge workers task in an office setting, these include writing essays and preparing presentations. Additionally, you may receive e-mail messages during the experiment. You are allowed to read these, as they may contain useful information. In directory XXX you can find some material that may be helpful for your tasks, or you can use the internet to find information.
The experiment is made up of three blocks of activities. Before each block we will present you with an instruction for the activities in that block. After each block you will be asked to fill out a small questionnaire. The entire experiment will take about 3 hours, depending on your own speed. The amount of compensation you will receive depends on how many tasks you finished and the quality of your work. The minimal amount you will receive is 30 euros, the maximum is 40 euros. In each block you will be asked to prepare 3 presentations. At the end of the experiment we will choose one of these for you to present to us."

Figure 4.1: The general instructions that were given to the workers during the experiment.

for the Information Retrieval community because it describes information seeking behaviour in a work context. In this chapter we describe how we collected and pre-processed the data and show statistics about the collected data. Additionally we will provide some examples of research that could be done with this dataset.

We will make the dataset available for research purposes.

4.2. METHOD

The main purpose of the data collection experiment that we carried out was to study stress among knowledge workers during a typical work day (Koldijk et al., 2013). The setup of the experiment was aimed at collecting data to recognize user activities during their work at the computer. The subjects were asked to write reports on a total of 6 given topics and prepare presentations for three of the topics.

The experiment in which the data were collected captured three conditions: a) a neutral condition in which the participants were asked to work as they normally do; b) a condition in which they were time pressured and c) a condition in which they were interrupted with email messages. Each of the conditions lasted between 30 and 45 minutes. In the remainder of this section we will describe the tasks in more detail. For more information on the conditions and the stress related data collection we refer to Koldijk et al. (2013).

4.2.1. PARTICIPANTS

We collected data from 25 participants with an average age of 25 (std 3.25). This number of participants is sufficient for within-subject measurements. There were 8 females and 17 males, and the participants were recruited among university students and interns at TNO[3]. 23 participants wrote their reports and presentations in English, two used Dutch. All participants received a standard subject fee for experiment participation. To motivate the students to do their best on the reports, they were told that the height of the fee was dependent on their performance.

4.2.2. MATERIALS

The participants executed their tasks on a laptop computer equipped with Microsoft Office. The default browser was Internet Explorer with www.google.nl as start page.

[3]Dutch institute for applied scientific research

Also, uLog version 3.2.5[4] was installed. uLog is a key-logging tool that saves the active application, window title, url or file that is open, caption information, mouse movements, mouse clicks and keyboard activity with timestamps. Additionally the desktop was recorded with GeoVision's CCS[5] and the browser history was saved with IEHistoryView version 1.65[6]. Additional data types include: camera recordings for facial expressions, heart rate, skin conductance and 3D postures using Kinect. The participants had access to the general instructions of the experiment at all times. These can be found in Figure 4.1.

TASKS

In each of the conditions the participants randomly received two out of six tasks, one opinion task and one task that required more information seeking. The short descriptions of the opinion tasks were:

- "Your experience and opinion about stress at work."
- "Your experience and opinion about healthy living."
- "Your experience and opinion about privacy on the internet."

The informational tasks were:

- "5 Tourist attractions in Perth, West Australia."
- "A plan for a coast to coast roadtrip in the USA."
- "The life of Napoleon."

The longer descriptions of the tasks as presented to the participants are available in the dataset.

E-MAIL MESSAGES

In the condition where participants were interrupted with email messages, these messages sometimes contained tasks or questions. This resulted in two additional topics in the data: Einstein and Information Overload. The exact content of the received email messages is available in the dataset.

QUESTIONNAIRE

We also collected responses to self-reporting questionnaires addressing Task Load, Mental Effort, Emotion and Perceived Stress. The participants were also asked about their interest in the topics and how complex the topic was to them. The outcome of this questionnaire will be discussed in a separate publication.

[4] http://www.noldus.com/human-behaviour-research/products/ulog
[5] http://www.geovision.com.tw
[6] http://www.nirsoft.net/utils/iehv.html

4.2.3. PROCEDURE

At the beginning of the experiment the participants were asked to fill out some questionnaires about their health background. They were also given the general instructions. At the beginning of each condition the participants were asked to watch a relaxing movie for 10 minutes. This was necessary to get an adequate resting heart rate baseline for the stress research. After that, the participants could look at the given topics for the condition and start their work. They were told to give a signal when they were ready in the neutral condition, or to stop writing when the timer went off in the time pressure condition. After the experiment leader paused all sensors, the participant was given a next set of questionnaires related to stress and a questionnaire related to interest in the topics. The participant was allowed to take a break and walk around between the conditions.

4.3. RESULTING DATASET

The dataset we present contains all computer interaction data that was recorded during the experiment. Most importantly this dataset contains the data coming from the uLog key-logger as well as the data collected from the IEHistoryView. Figure 4.2 presents a small excerpt from a uLog datafile. This example shows the *event* 'Window "http:// www.google.nl" – Windows Internet Explorer activated'.

```
</Event><Event>
  <EventType>Other</EventType>
  <EventAction>Window Activated</EventAction>
  <TimeStamp>2012-09-19T11:22:01.9745893Z</TimeStamp>
  <Date>2012-09-19</Date>
  <Time>11:22:01.9745893+02:00</Time>
  <Milliseconds>974</Milliseconds>
  <EventDescription>Window "http://www.google.nl/ -
  Windows Internet Explorer" activated.</EventDescription>
  <Control>
    <ControlType>window</ControlType>
    <ControlCaption>http://www.google.nl/ -
    Windows Internet Explorer</ControlCaption>
```

Figure 4.2: Example of uLog data

4.3.1. EVENT AGGREGATION AND MANUAL LABELLING

Besides the raw key-log data, we also provide a variant of the dataset in which the collected history-files and uLog-files have been preprocessed. In this dataset, individual events are aggregated to meaningful *event blocks*. We defined the start of a new event block as either an application switch, or a change in window title. In total, our data collection consists of 9416 event blocks.

All the keys typed and all captions (mouse-over tool tips) that occurred within one event block are concatenated into strings and the number of mouse clicks per event block is counted. From the recorded Google URLs we extracted the queries that were entered using a regular expression.

For future training and testing purposes (e.g. for topic detection), we collected

Figure 4.3: Example of presented event block to annotators. The typed text was shown in a box to the left of the file window

manual topic labels for the event blocks. For the labelling, we used the crowd sourcing platform Amazon Mechanical Turk because of the large number of event blocks. The event blocks were presented to the annotators in a desktop-like setting to mimic the desktop view of the user during the experiment (Figure 4.3). The annotators were asked to select 1 topic label and also indicate on a scale of 1-5 how certain they were of their decision. The event blocks were shown in random order, so they could not use any session information. The labels were the 8 topics, and an additional topic 'indeterminable' when the event block did not contain any identifiable topic, for example when just the website 'www.google.nl' was shown.

Table 4.1: Overview of features collected per event block, with example values

feature	example value
id	6
participant id	2
begin time	20120919T132206577
end time	20120919T132229650
duration (seconds)	24
# clicks	3
typed keys	we austra;i lia
application	iexplore
window title	Google - Windows Internet Explorer
caption	New Tab (Ctrl+T) New Tab (Ctrl+T)
url	http://www.google.nl/search?hl=nl& sclient=psy-ab&q=australia+&oq=australi
domain	www.google.nl
query	australia
Label	Perth

Table 4.1 shows an overview of the features collected per event block, with an example value for each feature. Table 4.2 shows the distribution of the labels in our data. 121 event blocks were not labelled because of problems with the annotating system (Amazon Mechanical Turk). Inter-annotator reliability was measured on a stratified subset of the data with 10 random event blocks from each participant. Cohen's κ was 0.784, which indicates a substantial agreement.

Table 4.2: Features in aggregated dataset and details on the labeled data

Total no. event blocks	9416
Average no. event blocks per participant	377
No. of 'Indeterminable' blocks	4347
No. of 'Einstein' blocks	117
No. of 'Information Overload' blocks	67
No. of 'Stress' blocks	612
No. of 'Healthy blocks	637
No. of 'Privacy' blocks	269
No. of 'Perth' blocks	1248
No. of 'Roadtrip' blocks	1170
No. of 'Napoleon' blocks	828
No. of blocks failed to label	121

4.3.2. EXAMPLES OF ANALYSES WITH THE DATA

Several types of analyses are possible with the data. In previous work (Koldijk et al., 2013), we have studied the relations between stressors (time pressure, incoming e-mails) and sensor measurements (number of keystrokes, facial expression). We now discuss two types of analyses related to Information Retrieval: a query-central (system-oriented) and a behaviour-central (user-oriented) analysis. The multimodal aspects (camera and desktop recordings) of the data may be exploited in future research.

SYSTEM-ORIENTED ANALYSIS

As an example of a system-oriented analysis, we investigated the automatic estimation of relevance of URLs for each user query based on the user's interactions following the query. Per query, we extracted the URLs that were accessed in the query's event block and the event block after it. A common variable to estimate the relevance of a page is a dwell time of at least 30 seconds (Guo and Agichtein, 2012). Since we have interaction data available, and we know that the user was collecting data for writing a report or a presentation, we not only calculated the dwell time on each web page but also registered whether the next active application was Word or Powerpoint, and if the user typed control-c in the browser before making this switch (copying text). Table 4.3 shows the results of this analysis. Note that the total number of queries in the first row includes duplicate queries that are recorded when the user clicks on a result and goes back in his browser. The table shows that if dwell time is used as only relevance criteria, only 44 pages would be judged as relevant. Taking

Table 4.3: Results of the query-central analysis

Total # of queries	980
of which followed by a click on a URL	732
of which followed by a switch to Word/Powerpoint	125
of which with control-c	15
with a dwell time of >= 30 seconds	44

into account the switches to Word and Powerpoint, this number is much higher. Table 4.4 shows an excerpt of the processed interaction data, focussing on a series of queries and the clicked web pages for those queries.

BEHAVIOURAL ANALYSIS

Figure 4.4 shows an example of a behavioural analysis: a transition graph for the workers' information behaviour. It shows that when users are asked to write reports or prepare presentations on a relatively new topic, they spend more time on web pages than in the report they are writing, and they switch frequently between URLs and the report in order to gather the relevant information. The graph also shows the relatively frequent interruptions of e-mail, which is known to be very common for knowledge workers (Whittaker and Sidner, 1996).

4.4. DISCUSSION

We encountered a few challenges in processing and analysing our dataset. First, the data that we collected is rich and comes from multiple sources. We found that combining the data from the key-logging software and the browser history software was not trivial, even with exactly matching timestamps. This was because the user could have multiple tabs active in the browser, with not all tab titles being separately recorded by the key-logging software. Second, users clicking one of Google's query suggestions sometimes led to incomplete queries and missing URLs. For example, we found that the query 'napol' lead to the URL http://nl.wikipedia.org/wiki/Napoleon_Dynamite. We suspect that his happened because the user selected the suggested query 'napoleon dynamite' after the offset 'napol', and then clicked the Wikipedia URL. Third, in some cases the window title of the browser did not change when a user clicked on a result (especially when the click was a result from Google Images), which caused the clicked URL to be included in the same event block as the query, and dwell time was missing for this particular URL. Fourth, the browser logging resulted in a lot of noise. We had to filter out a large amount of on-page social media plug-ins, advertisements and icons. In addition, browsing the Google domain leads to many additional URLs. An extreme example was 25 occurrences of the Google Maps URL http://maps.google.nl/maps?hl=nl &q=usa&bav=on.2 in one event block.

Table 4.4: Excerpt of the processed interaction data, focussing on a series of queries and the clicked web pages for those queries. Note the spelling errors in the queries that apparently not lead to information finding problems. The URL http://cm.g.doubleclick.net/push illustrates that in some cases, noise from advertisement was logged instead of the containing web page.

Query	Clicked URL	Time on page	Key	next application
the life of napole	http://en.wikipedia.org/wiki/Napoleon	19 seconds		iexplore
the life of napoleon bonaparte	http://en.wikipedia.org/wiki/File:Napoleon_in_His_Study.jpg	8 seconds	CTRL+C	WINWORD
how to write a biography	http://homeworktips.about.com/od/biography/a/bio.htm	31 seconds		WINWORD
facts about napoleion	http://www.sheppardsoftware.com/Europeweb/factfile/Unique-facts-Europe10.htm	11 seconds		iexplore
facts about napoleon bonaparte	http://cm.g.doubleclick.net/push	7 seconds		WINWORD
family tree napoleon bonaparte	http://www.genery.com/sites/all/themes/gen2/images/screen/bonaparte.png	3 seconds		iexplore

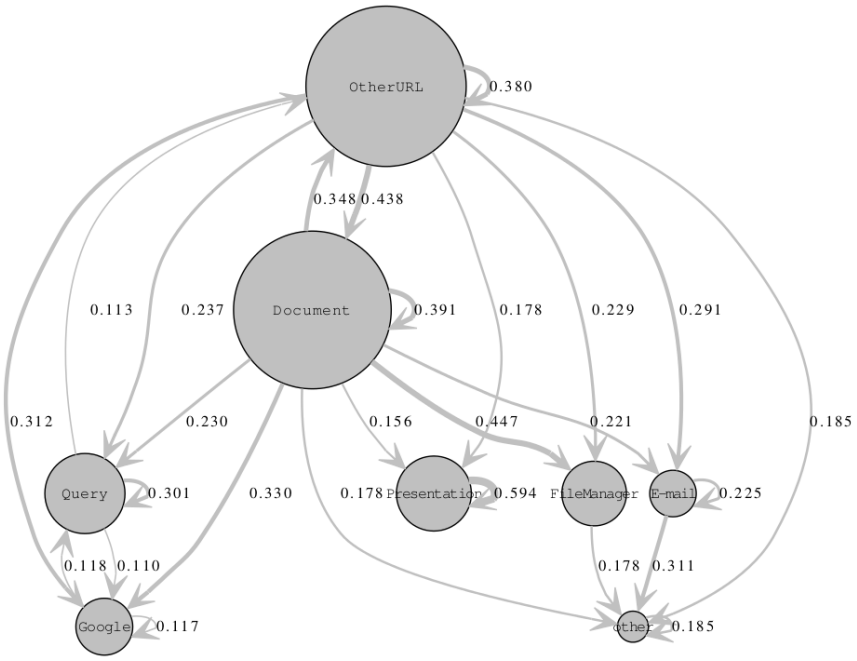


Figure 4.4: Transition graph for the information behaviour of the workers. The size of the states represents their relative frequency. The state 'query' represents events where the active application is the web browser, in which a Google query and its results are shown. The state 'Google' represents events where a Google page is active without query. The state 'OtherURL' represents events where the active application is the web browser, with another URL than Google. The transition probability between states S_1 and S_2 was calculated as $count(S_1 \rightarrow S_2)/count(S_2)$. Only transitions with a probability > 0.1 are shown.

4.5. CONCLUSIONS AND FUTURE WORK

We collected and preprocessed a dataset of information behaviour of knowledge workers in the context of realistic work tasks. The data set is relatively small in terms of the number of participants, but large in terms of types of information collected. The contributions of this dataset are: 1) it includes different types of data, including key-logging data, desktop video recordings and browser history; 2) the information seeking behaviour is completely natural because it results from the recording of user behaviour during report writing, e-mail reading and presentation preparation; and 3) the search activities have been recorded together with the context of these other tasks, which allows for future research in context-aware information retrieval.

II

CONTEXT OF THE KNOWLEDGE WORKER

For me context is the key –
from that comes the understanding of everything.

Kenneth Noland

5

THE KNOWLEDGE WORKER AND HIS CONTEXT

In this chapter we give an introduction on the notion of context. Furthermore we present a conceptual and formal model of the context of a knowledge worker. The context of a knowledge worker is highly dynamic and driven by events. We assume that the knowledge worker is the centre of the context, but can be influenced by the context as well. We assume that we can observe the context using sensors, independent from the knowledge worker, but know that the interpretation of the sensed elements is dependent on the knowledge worker.

The model includes the notion of a knowledge worker, resources and a possible interaction between them. Primarily the knowledge worker is engaged with a task (part of the knowledge worker context) and the user can be distracted or helped by interaction with resources.

5.1. INTRODUCTION

An important concept that we use in this thesis is the notion of 'context'. 'Context' is a concept that is often used, but rarely defined. Since this is a possible source for miscommunication, we provide some background on context and how we interpret context in the remainder of this chapter.

We start with an overview of more philosophical literature on the concept of 'context' in Section 5.2. Then we zoom in on the knowledge worker and his specific context in order to answer the question "How can we model the context of a user and what are the requirements of this model?". We present a conceptual model of the knowledge worker model in Section 5.3 and formalize it in Section 5.4. We end with some possible application scenarios for the model in Section 5.5.

5.2. DEFINITION OF CONTEXT

Previous attempts to get a grip on the concept of context describe it as a continuous scale where each end of the scale describes an interpretation of context. Usually no final definition is given, only a preference on where to reside on the scale. In the literature, multiple dimensions for defining context have been proposed. We describe the three that we consider most important:

- Container vs. meaning (Dervin, 1997)
- Objective vs. subjective (Penco, 1999)
- Interactional vs. representational (Dourish, 2004)

5.2.1. CONTAINER VS. MEANING

Dervin (1997) describes context as a continuum where at one extreme of the continuum context is interpreted as "a container in which a phenomenon resides": "context has the potential of being virtually anything that is not defined as the phenomenon of interest" (p.112). So for example, if you receive a postcard everything but the postcard itself can be considered as context; the envelop, where it was posted, the occasion for sending it, but also the mail-man that delivered it, the vehicle that delivered it etc. These choices are endless. Therefore, in practice dealing with context is a matter of choosing which aspects of the container relate to the phenomenon and which aspects are relevant for your goal.

At the other extreme of the continuum, context is interpreted as "carrier of information": "context is assumed to be a kind of inextractible surround without which any possible understanding of human behaviour becomes impossible" (p. 113). In this interpretation context is still centred around a phenomenon, but it is also intertwined with the phenomenon; context gives meaning to the phenomenon and the phenomenon gives meaning to the context and it is difficult to see where context begins and the phenomenon ends. Dervin finds that in this sense, every context is by definition different, and cannot be generalized. For applications it is rather impossible to use the interpretation where context is the carrier of information, at best an application can use context in a way where it adds meaning to a situation.

5.2.2. OBJECTIVE VS. SUBJECTIVE

A second continuum for context interpretation is given by Penco (1999). He distinguishes the objective view on context; a metaphysical state of affairs, where context is simply a set of features of the world, from the subjective one, in which context is seen as a cognitive representation of the world; a set of assumptions about the world (e.g. beliefs). He concludes that you can never reach a definite representation of context as it is assumed to be endless. In the end Penco doubts the existence of a true objective context, since he beliefs that even the objective features of the world are given from some subjective point of view. In most applications features will be acquired in a consistent way, favouring the objective interpretation of context.

5.2.3. INTERACTIONAL VS. REPRESENTATIONAL

The continuum by Dourish (2004) compares a representational view to an interactional view. The representational view is explained by the definition of Dey, Abowd, and Salber (2001): "any information that characterizes a situation related to the interaction between humans, applications, and the surrounding environment" (p.97). Dourish observes that this definition assumes that context can be known (context is information), that one can define in advance what is included as context (context is delineable), that the relevance of potential contextual elements is always the same (context is stable) and that activity happens within a context (context is separable from the activity). With this interpretation, practical use of context in applications becomes a matter of finding which aspects should impact the behaviour of the application. On the other end of the continuum Dourish positions the interactional view. This view assumes that context is a relational property, meaning that something might be contextually relevant, but it is not context in itself. This also means that context cannot be defined in advance, but has to be defined dynamically. In its turn this implies that context is unique for each occasion of an activity, and thus that context arises from the activity. Now the problem for context-aware applications becomes a matter of finding mutual understanding of context between the user and the application rather than the proper encoding of context.

5.2.4. CONTEXT IN PERSONAL INFORMATION MANAGEMENT

All of these descriptions of context seem intuitive to some extent. There is a strong relation between the container-view by Dervin, the objective view by Penco and the representational view by Dourish in the sense that they all state that the context is merely a set of information. However, the other extremes of the scales vary quite a bit. We conclude that it is impossible to give a single definition of context. After all, the definition of context itself is context-dependent. We can only sketch where our interpretation resides on each of the scales and why. Of course this motivation, and thus our choice of how we interpret context is context-dependent. In our case it is determined by our end-goal, our final application, a system that supports working in context.

Other researchers also took an application area as starting point for an operational definition of context. We describe a few of these approaches in terms of the three scales that we described earlier.

In the personal information management scenario, Gomez-Perez et al. (2009) define context as "a set of information objects that are frequently accessed concurrently or within a very short time-span". Additionally, information objects that are similar in terms of content may belong to the same context as well. They stress that for "working in context" to be helpful, relations between information sources in the same context need to be meaningful to the knowledge worker and therefore they leave the actual definition of context (e.g. which groupings of objects are relevant) to the user. Since the context is defined by the user we interpret this as a subjective view on context. The feature-based approach and reasoning from information objects indicates a representational interpretation, but the fact that they consider irrelevant groupings of information objects to be meaningless suggests a view towards towards "context

as meaning" on the scale of Dervin.

In contrast, several researchers have adopted the objective, representational, "context as container" view (Ermolayev et al., 2010; Whiting and Jose, 2011; Devaurs, Rath, and Lindstaedt, 2012): The ontological context model for knowledge workers by Ermolayev et al. (2010) is based on a pragmatic selection of things that are related to the entity on which the context is about. These include processes (for example development of a controller) and objects such as persons, resources, tools etc. Whiting and Jose (2011) share this view. Their model provides contextualized recommendations of previously accessed information sources. The contextual elements that are used for recommendation are summarized. These are fixed and measured independent from the user's beliefs.

Devaurs, Rath, and Lindstaedt (2012) also seem to agree with this view. They present an ontological model for context, but do not take into account the user. Ingwersen and Järvelin (2005) describe their nested model of context stratification (p.281) in which they take the container-view one step further. Here context is centralized around a core with multiple dimensions of context around this core. In a sense, these dimensions are all nested containers. Additionally, the core can be either an object or a person, suggesting the possibility for both a subjective and an objective view on context. There is also an emphasis on actor's experiences that form expectations, proposing a somewhat interactional view. In Figure 5.1 these practical views on context in the personal information management and recommendation scenario are mapped to the dimensions by Dervin, Penco and Dourish.

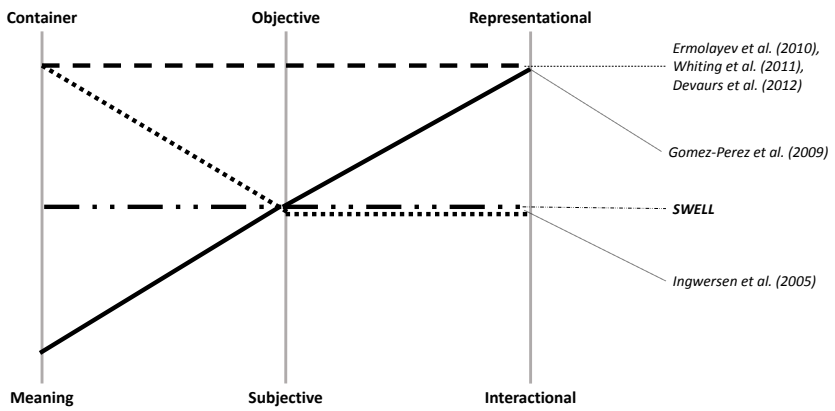


Figure 5.1: Visualization of the scales of context and where our interpretation and approach to context resides compared to a selection of existing approaches in the domain of personal information management and recommendation

We propose to take the golden mean in terms of these scales. We assume that this golden mean will make an application more robust as it will have elements of all

views. In our proposal context is centred around a user, consistent with the container view, but the context will also be used as something that adds meaning to a situation. The context will be sensed in an objective setting, but the user will be taken into account in weighting the sensed elements, which adds a subjective component. Finally some contextual elements will be defined of which we think are relevant for the application (representational), but we reason based on events (interactional). Additionally we distinguish *detecting* context, which is highly dynamic (interactional), from *identifying* context (an aggregated form of context) which is more representational in nature.

As the definition of context is vague, most researchers in context-aware systems define their own set of contextual elements that they model. Since there is no consensus on the contextual elements to include we present our own view on context in this sections. We focus on the contextual elements of the knowledge worker's life that are relevant for supporting him in order to make his work more effective and efficient.

5.3. CONCEPTUAL MODEL OF THE KNOWLEDGE WORKER'S CONTEXT

When we talk about the context of a knowledge worker, the user (the knowledge worker) is the central point of reasoning, and we are interested in those elements that influence the user. However, it is still unclear which elements are important and how they interact with each other. In Figure 5.2 we present our view on the knowledge worker and his context in a conceptual model[1]. This model is formalized in Section 5.4. The formal model can be used to reason about certain activities of the knowledge worker, such as which resource he is going to select, what the knowledge worker is going to learn, and how we can determine which task a knowledge worker is executing.

We consider a knowledge work environment. Typical for such an environment is that it includes a knowledge worker and one or more resources. The knowledge worker interacts with these resources to achieve his or her goals. Goals are achieved by formulating strategies, which consist of one or more tasks. We assume that a knowledge worker is a person who is characterized by:

- a task that the knowledge worker wants to execute, which is part of a strategy that is devised to achieve some goals related to knowledge work.
- a cognitive state: consisting of (1) general knowledge that the knowledge worker has obtained by education or by in previous experiences and (2) assumptions about the knowledge work environment in general and the resources in it in particular.
- an emotional state: the emotions and energy a knowledge worker has

[1] Edited from: **Maya Sappelli, Suzan Verberne, Wessel Kraaij** (2015) *Adapting the interactive activation model for context recognition and identification*, Submitted to: ACM Transactions on Interactive Intelligent Systems.

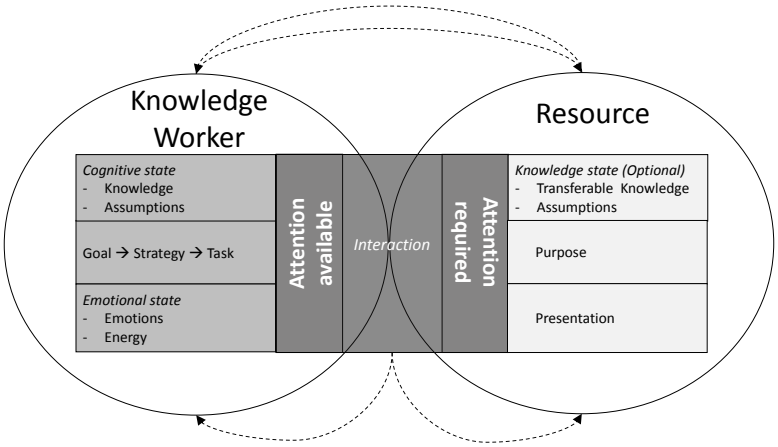


Figure 5.2: Conceptual model of context for a knowledge worker.

- at each moment a limited amount of available attention that he can give

We assume that a resource is characterized by:

- a purpose for which it was created
- a knowledge state (optional): consisting of (1) knowledge that can be transferred and (2) in the case of personalized systems, assumptions about the knowledge worker
- presentation (in what condition is the resource). The presentation is assumed to contribute to both formal and informal communication
- at each moment a required amount of attention that is needed for successful interaction

Examples of resources are: another person, a device such as a computer, but also the lighting in the location where the knowledge worker is. Of course a resource such as a lamp has no assumptions and does not have relevant knowledge that it can transfer. A lamp, however, is capable of influencing the attention and emotions of a knowledge worker, which is why it is still considered as a resource.

In the context of the task to be executed, knowledge worker K may choose to interact with some resource R. For this interaction we assume:

- The interaction of K with R has an effect on both the involved knowledge worker and resource, but not on other knowledge workers or resources.
- The interaction of the knowledge worker with the resource is rationally guided by the task that K wants to execute. Furthermore, the interaction is constrained by the assumptions and knowledge K has of resource R.

- Additionally the interaction is constrained by the purpose for which resource R was created.
- The interaction of K with R can be (positively or negatively) influenced by the emotions of K.

There are two possible effects from the interaction of K with R:

- the intended effect: the effect that was anticipated by K, and was the reason to initiate this interaction. Most likely this effect brings K closer to fulfilling his or her task.
- the unintended effect: the effect that was not anticipated by K.

These interactions are influenced by the amount of attention that K has available. Some things will not be perceived consciously, because K is not focusing on it. This creates two possibilities:

- K is focused on the effects of the interaction that help to achieve the tasks involved.
- K is focused on the effects of the interaction that distract from these tasks.

Furthermore, a resource R can have a conscious or unconscious effect on the knowledge worker, even when he or she does not explicitly interact with R. This effect can be on all aspects of K. For example the presentation of R can alter the assumptions K has about R (form example, R looks broken to K). Also, it can have an effect on the emotional state of K or the amount of available attention.

In this thesis we assume that the knowledge worker takes the initiative to interact with a resource, and that this can not happen the other way around. Furthermore we do not consider the conscious or unconscious effects that a resource may have on a knowledge worker when there is no explicit interaction.

To achieve his goals, a strategy of knowledge worker K could be to engage in the interaction with a resource that most likely has the best positive influence on the current task.

5.4. FORMAL MODEL OF THE KNOWLEDGE WORKER'S CONTEXT

In this section we extract the relevant concepts from the way of thinking and describe their relations. A knowledge work environment $\mathsf{Kwe} = \langle \mathsf{Kw}, \mathsf{Rs}, \mathsf{Kd} \rangle$ is modelled as a structure consisting of:

1. a set Kw of so-called knowledge workers,
2. a set Rs of so-called resources,
3. Kd is a knowledge domain, consisting of sets Kn of knowledge items.

In our model, we see a knowledge state as the assignment of an *awareness* level (real-value taken from $[0,1]$) to each knowledge item. The set Ks of knowledge states thus is described as $\mathsf{Ks} = \mathsf{Kd} \to [0,1]$, or, the set of all such functions. There are no limitations as to what a knowledge item can be.

Furthermore, we assume a set E_k of emotional states that knowledge workers may have (e.g. happy, sad), and a set E_r describing the presentation states that a resource may have (e.g. functioning, broken).

The set $\mathscr{T}$ consists of all possible tasks some knowledge worker may perform. Each task $t \in \mathscr{T}$ is seen as a structure $t = \langle \mathsf{N}, \mathsf{L} \rangle$ where N is the identifying task label and $\mathsf{L} \in \mathsf{Ks}$ the required knowledge state to be able to fulfil that task.

Next we focus on the knowledge workers. A knowledge worker $K \in \mathsf{Kw}$ is a structure $K = \langle \mathsf{T}, \mathsf{L}, \mathsf{E}, \mathsf{F} \rangle$ consisting of:

1. a set $\mathsf{T} \subseteq \mathscr{T}$ a tasks to be performed by that knowledge worker,
2. the knowledge state $\mathsf{L} \in \mathsf{Ks}$ of the knowledge worker;
3. the current emotional state $\mathsf{E} \in \mathsf{E}_k$ of the knowledge worker;
4. the attention/focus $\mathsf{F} \in \mathsf{Kd}$ of the knowledge worker.

The set $\{k \in \mathsf{Kd} \mid \mathsf{F}(k) > \theta_1\}$ consists of all knowledge items that are in the focus of the knowledge worker and which can be learned, where θ_1 determines the threshold for knowledge items to be in that learning set.

We continue with the discussion of resources. In our model, a resource R from Rs is seen as a structure $R = \langle \mathsf{P}, \mathsf{L}, \mathsf{E}, \mathsf{F} \rangle$ consisting of:

1. a set $\mathsf{P} \subseteq \mathscr{T}$ of tasks for which the resource can be used,
2. the current knowledge state L of the resource
3. the current presentation state $\mathsf{E} \in \mathsf{E}_r$ of the resource
4. the attention/focus $\mathsf{F} \in \mathsf{Ks}$ of the resource.

Only the items that are sufficiently visible in the focus F will be transferred to the knowledge worker who requested the interaction with the resource. A knowledge item r is sufficiently visible in the focus, when its awareness exceeds some threshold θ_2, or: $\mathsf{F}(r) > \theta_2$. An example of the focus of a resource such as a computer are the knowledge items that are visible on the screen.

This completes the introduction of the concepts and their relations in our model.

5.4.1. OPERATORS ON KNOWLEDGE STATES

In this subsection we will introduce some convenient operators on knowledge states.

commonality The commonality $f \odot g$ of knowledge states f and g is defined as: $(f \odot g)(x) = \min(f(x), g(x))$ for all $x \in \mathsf{Kd}$.

aggregation Let f and g be two knowledge states, then the aggregation $f + g$ of f and g accumulates these both knowledge states into a new one. The operator $+ : \mathsf{Ks} \times \mathsf{Ks} \to \mathsf{Ks}$ is defined as: $(f+g)(x) = \min(f(x) + g(x), 1)$ for all $x \in \mathsf{Kd}$.

difference The difference $f - g$ of knowledge states f and f is is defined by $(f - g)(x) = \max(f(x) - g(x), 0)$ for all $x \in \mathsf{Kd}$. For convenience, we assume the operator $-$ is left-associative, meaning that $f - g - h$ is to be interpreted as $(f - g) - h$. Furthermore, is is easily derived that $f - g - h = f - h - g$.

similarity The function $\mathsf{Sim} : \mathsf{Ks} \times \mathsf{Ks} \rightarrow [0, 1]$ measures the similarity between knowledge states. A typical definition is the so-called cosine measure.

5.5. USING THE MODEL OF THE KNOWLEDGE WORKER'S CONTEXT

The concepts and their relations as introduced in the previous subsection, allow us to reason about our application domain. We will describe a typical interaction step of a knowledge worker k. Note that as a result of this interaction, the internal state of the knowledge worker will change. We assume that the knowledge worker is involved in task t. Consequently, the knowledge worker has to extend his knowledge state from L_k to cover the required knowledge L_t also, or, $\mathsf{L}_k + \mathsf{L}_t$. Usually this will require interaction with several resources. We will describe how the knowledge worker may select the best resource and what the effect of the interaction on the knowledge worker will be. Figure 5.3 can be used as a visual representation of the knowledge domain and the relations that play a role in these selection and learning processes.

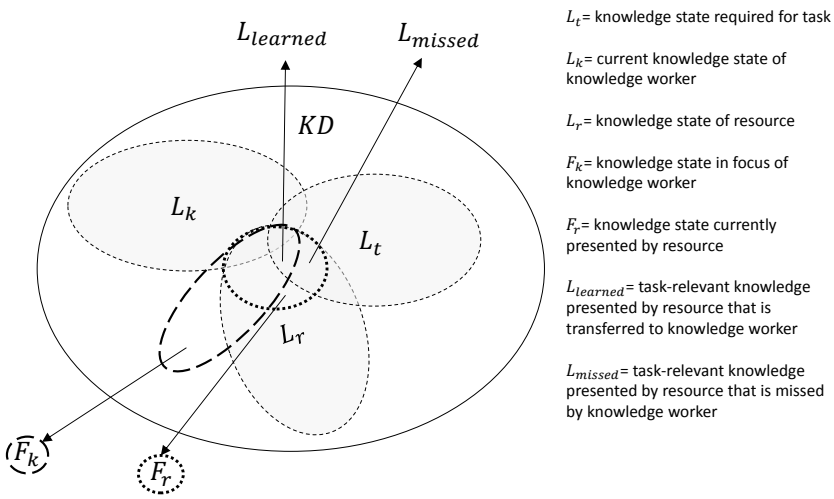


Figure 5.3: Visual representation of the knowledge domain and relations between knowledge state concepts

The knowledge gap may be expressed as $\mathsf{L}_t - \mathsf{L}_k$.

As a result, the knowledge worker will select the resource r with knowledge provision potential L_r most similar to this knowledge gap:

$$r \in \mathsf{argmax}_x \mathsf{Sim}(\mathsf{L}_x, (\mathsf{L}_t - \mathsf{L}_k)$$

where $\mathsf{argmax}_x f(x)$ is the set of all arguments x for which $f(x)$ is maximal.

Next we consider the situation after the interaction of the knowledge worker with this resource r. As a consequence of the input provided by the knowledge worker during the interaction, we assume the resource r has presented F_r to the knowledge worker. So the knowledge worker could in theory have learned $\mathsf{L}_{possible} = (\mathsf{F}_r - \mathsf{L}_k)$ from this interaction with r. However, we assume that the learning takes place in the context of a task. This means that we can further categorize the learning gain in an unintended and an intended part:

1. The intended gain is what the knowledge worker wanted to learn to come closer to the fulfilment of task t: $\mathsf{L}_{learned} = (\mathsf{L}_t - \mathsf{L}_{possible}) \odot \mathsf{F}_k$. Since the learning is restricted to this focus F_k of the knowledge worker it is possible that the intended gain was not optimal. The remaining part is what the knowledge worker could have learned but missed: $\mathsf{L}_{missed} = (\mathsf{L}_t - \mathsf{L}_{possible}) - \mathsf{F}_k$.

2. The unintended gain is that what the knowledge worker learned but was not related to the task: $\mathsf{L}_{extra} = \mathsf{F}_k - \mathsf{L}_{possible}$. The unintended gain can come from the resource with which the knowledge worker was interacting, or from the resources that he was not consciously interacting with:
   (a) unintended gain from resource r: $\mathsf{L}_{extra} = (\mathsf{F}_k - \mathsf{L}_{learned}) \odot \mathsf{F}_r$
   (b) unintended gain from remaining resources $\mathsf{L}_{extra} = \mathsf{F}_k - \mathsf{F}_r$

After the interaction with the resource, knowledge worker k will be in the state $\mathsf{L}_r + (\mathsf{F}_r - \mathsf{L}_k)$. At this point the focus of the knowledge worker and / or emotional state may have changed, and, if required, the knowledge worker is ready to select the next resource for interaction.

The model can also be used to deduce which task a knowledge worker is most likely working on, which is a form of context *identification*. Task or context *prediction* is useful for many context-aware support mechanisms such as the one we describe in Chapter 9. In order to predict which task a knowledge worker is working on, we need to observe the information that the knowledge worker is focused on F_k. Assuming that the knowledge worker will focus on the information that is relevant for his task, the most likely task $t \in T$ that the knowledge worker is engaged in is the one for which the similarity to F_k is largest: $t = \mathsf{argmax}_t \mathsf{Sim}(L_t, F_k))$. In the next chapter, we operationalize the model presented in this chapter in order to predict which task, or more specifically which project, a knowledge worker is working on.

5.6. CONCLUSION

In this chapter we gave an introduction to the notion of context. Inspired by the literature in Section 5.2 we conclude that the context of a knowledge worker is highly dynamic and driven by events. We assume that the knowledge worker is the centre of the context, but the context can influence the knowledge worker as well. We assume that we can observe the context using sensors, independent from the knowledge worker, but know that the interpretation of the sensed elements is dependent on the knowledge worker.

Another important conclusion of the literature was that in order to make it more clear what is meant by "context", a subset of contextual elements that is relevant for the application needs to be described. Therefore, we described the context of a knowledge worker in more detail in a conceptual and formal model.

In the conceptual model of the context of a knowledge worker we focus on the interaction of the knowledge worker with his surroundings. Both the knowledge worker as well as the resources he interacts with are partly observable and partly unobservable.

The formal model showed us that the model is indeed capable of modelling the aspects that are important in a knowledge worker's life. Using the model it is possible to reason about resource selection (which resource will the knowledge worker interact with) and learning (which knowledge will the knowledge worker gain at a certain moment in time). The model also allows for reasoning about which task a knowledge worker is most likely involved in at a certain moment. This is a form of context *identification*, where the context is aggregated into a task and is useful in context-aware applications. In the next chapter we will operationalize the context model in order to do evaluate its effectiveness in context *identification*. In Chapter 9 we will use the operationalized model for context-aware support in the domain of document recommendation.

6

ADAPTING THE INTERACTIVE ACTIVATION MODEL FOR CONTEXT RECOGNITION AND IDENTIFICATION

Edited from: **Maya Sappelli, Suzan Verberne, Wessel Kraaij** (2016) *Adapting the interactive activation model for context recognition and identification*, Under revision: ACM Transactions on Interactive Intelligent Systems.

In this chapter, we propose and implement a new model for context recognition and identification. *Our work is motivated by the importance of 'working in context' for knowledge workers to stay focused and productive. A computer application that can identify the current context in which the knowledge worker is working can (among other things) provide the worker with contextual support, e.g. by suggesting relevant information sources, or give an overview of how he spent his time during the day.*

Our model is based on the interactive activation model. It consists of a layered connected network through which activation flows. We have tested the model in a context identification setting. In this case the data that we use as input is low-level computer interaction logging data.

We found that topical information and entities were the most relevant types of information for context identification. Overall the proposed model is more effective than traditional supervised methods in identifying the active context from sparse input data, with less labelled training data.

6.1. INTRODUCTION

In our project SWELL[1] we aim to support knowledge workers in their daily life. One aspect is their working life. With the increasing amount of information they have to handle, knowledge workers can get overwhelmed easily: a phenomenon referred to as 'information overload' (Bawden and Robinson, 2009), and filtering irrelevant information or 'working in context' is deemed beneficial (Gomez-Perez et al., 2009; Warren, 2013). Additionally, with the arrival of smart phones and "any place, any time information" (e.g. the wish and opportunity to access information at any place at any time), proper work-life balance is at risk.

This creates two use cases for supporting knowledge workers. In the first use case ('working in context') we aim to support knowledge workers by filtering information based on their current activities (providing contextual support). Ardissono and Bosio (2012) have found that task-based and context-based filtering reduce the user's current workload. Thus, by recommending and highlighting information that is relevant to the context, while blocking information that is out-of-context, we help the user to stay focused on his current task.

In the second use case ('user-context awareness') we aim to make users aware of their activities and work-life balance, by showing them a record of their activities. A concrete example is by means of 'hour tracking'. Many companies ask their employees to define how much time they spend on each project during a week for cost definition purposes. By providing the user with an automatic overview of his day or week, the employee can save time on this task.

Both use cases, that are closely related to life logging (Gurrin, Smeaton, and Doherty, 2014), require us to keep track of what the knowledge worker is doing during the day. That is, we aim to identify the user's *context*.

Unfortunately, context is a vague concept. Many researchers (Akman and Surav, 1996; Dervin, 1997; Penco, 1999; Dey, Abowd, and Salber, 2001; Dourish, 2004) have tried to define the concept, but it seems difficult to get a good grip on it. There is neither a clear answer to what the 'context' in 'working in context' is nor how it can be recognized automatically. This will be the focus of this chapter.

In section 5.2 we analysed related literature on context and context modelling. We are not only interested in what context should look like for our application, but also how the user's activities can be mapped to meaningful contexts. That is why we present an overview of existing approaches to recognizing context automatically in section 6.2.2. One of the conclusions is that a complete conceptual framework for describing and reasoning about context and its constituting elements is missing. We presented our own definition and model of context in section 5.3. These provide the starting point for a computational model of context. The main contribution of this chapter is a novel approach and implementation for context *recognition* and *identification* which is described in section 6.3. Compared to existing approaches, this method aims to keep the effort to use the system as low as possible. This means that little or no labelled data is required to initialize the method, which ensures that we do not add to the load of the knowledge worker.

[1] http://www.swell-project.net

Our research questions are:

1. RQ1: How can we implement the model for context identification in a way that requires a minimal amount of labelled data for training?

2. RQ2: What information is required for successful context identification?

3. RQ3: How effective is our model in identifying the user's context?

6.2. BACKGROUND AND RELATED WORK

In this section we present an overview of literature on context in personal information management and context recognition and identification approaches.

6.2.1. CONTEXT IN PERSONAL INFORMATION MANAGEMENT

Context is a concept that is often used, but rarely defined. Since this is a possible source for miscommunication, we provide some background on context and how we interpret context in the remainder of this chapter. We describe literature from the field of personal information management (a sub field of Information Retrieval and Information Science), as this area is most relevant for the support of knowledge workers.

In the research area of personal information management, Gomez-Perez et al. (2009) define context as "a set of information objects that are frequently accessed concurrently or within a very short time-span". Additionally, information objects that are similar in terms of content may belong to the same context as well. They stress that for "working in context" to be helpful, relations between information sources in the same context need to be meaningful to the knowledge worker and therefore they leave the actual definition of context (e.g. which groupings of objects are relevant) to the user.

In contrast, several researchers have adopted a view on context that is not dependent on the personal interpretation of the user (Ermolayev et al., 2010; Whiting and Jose, 2011; Devaurs, Rath, and Lindstaedt, 2012): The ontological context model for knowledge workers by Ermolayev et al. (2010) is based on a pragmatic selection of things that are related to the entity on which the context is about. These include processes (for example development of a controller) and objects such as persons, resources, tools etc. Whiting and Jose (2011) share this view. They attempt to provide contextualized recommendations of previously accessed information sources and summarize the contextual elements they use for that purpose. These are fixed and measured independent from the user's beliefs.

Devaurs, Rath, and Lindstaedt (2012) also seem to agree with this view. They present an ontological model for context, but do not take into account the user. Ingwersen and Järvelin (2005) describe their nested model of context stratification (p281) in which they centralize context around a core and see multiple dimensions of context around this core. In a sense these dimensions are all nested containers. Additionally, the core can be either an object or a person, suggesting the possibility for both a subjective and an objective view on context.

In our approach, we will centre context around a user, allowing a subjective interpretation of what this context should entail. We will sense context in an objective setting, independent of the user. However, we will take the user's actions into account in determining the importance of the sensed elements, to maintain a subjective focus. In order to do so we will define some contextual elements of which we think are relevant for the application, similar to previous approaches. Since we determine the active contextual elements based on sensed events, the context detection becomes highly dynamic.

6.2.2. CONTEXT RECOGNITION AND IDENTIFICATION

In the previous section we summarized some literature on the concept of context from the field of Personal Information Management. Now we review the literature into the process of automatic context recognition as it is an important element in context-aware personal information management systems. In the presented literature, context recognition is essentially the mapping of one or multiple events (such as the user's active windows and typed keys) to a label that can be interpreted by the user as a meaningful activity. We actually see this as context *identification* rather than *recognition*. From the literature on context we have learned that context usually entails a collection of many elements, thus context recognition should be the recognition of these elements, and not the process of summarizing these elements in a communicable label. In the remainder of this chapter we will use context identification when we refer to the representation of context by a single label, while we will use context recognition when we describe context by all its elements.

In any case, the context identification methods presented in literature vary in the interpretation of what type of identification is interesting. The different types of context identification methods we look at are: topic-based (section 6.2.2.1), process-based (section 6.2.2.2) and memory-based (section 6.2.2.3). We present our own approach in section 6.3.

TOPIC-BASED CONTEXT IDENTIFICATION

The methods for topic-based context identification focus on generalizing events to topics. For example by identifying some computer activities as related to "trip to Rome" vs. "trip to Paris".

There are several approaches. A first group of studies essentially sees context identification as a *categorization* problem. These approaches are similar to document categorization as they typically monitor the terms in the documents, document sequence, or window title and map them to one of the context categories. The classification algorithm varies from network-based (WordSieve by Bauer and Leake (2001)), graph-based (SeeTrieve by Gyllstrom and Soules (2008)), Bayesian classifiers (IRIS by Cheyer, Park, and Giuli (2005) and TaskTracer by Stumpf et al. (2005)) to SVM (Task Predictor by Shen et al. (2006)).

Secondly, there are approaches based on clustering where the process is mainly about finding clusters of related documents or windows and evaluating these on labelled data. In the Swish-system by Oliver et al. (2006) windows are clustered using latent semantic indexing, in ACTIVE (Warren et al. (2010) and Štajner, Mladenić, and

Grobelnik (2010)) the authors use a weighted sum of cosine similarity for term overlap, social-network overlap and temporal proximity of documents and document access. In ACTIVE the document or information object is central; context identification is simplified to recording the cluster-tag that is given to the active information object at cluster-time.

In a third approach by Maus et al. (2011), context identification is primarily a manual process, done in their system Contask, which is integrated in the 'Nepomuk Semantic Desktop' (Groza et al. (2007)). Users define tasks and can associate information objects with these tasks. The users themselves are responsible for maintaining the appropriate active context thread, however Contask does provide a service where context switches are automatically detected with the purpose to propose the user to initiate a context switch.

The reported accuracies and precision-recall values are difficult to compare as each author evaluated his algorithm on small and private datasets. There is no publicly available dataset to compare results because of privacy concerns related to the data.

In these works, the main source of information is document content. In our work we propose to use keystrokes, mouse clicks and window information as well as the content of documents and other information objects as input variables for context recognition and identification.

PROCESS-BASED CONTEXT IDENTIFICATION

In this section we describe literature on process-based context identification methods. These focus on identifying a context by generalizing the process that is involved in the example. For example by identifying some activities as "planning a trip" vs. "claim expenses". Compared to the topic-based approaches the classes vary in the process that is involved, rather than the subject of the activity as was the case in "trip to Rome" vs. "trip to Paris", which would both be classified as "planning a trip" in the process-based approach.

For this type of context identification the approaches are similar to the topic-based approaches (6.2.2.1). Granitzer et al. (2009) use a traditional classification approach in which they compare the performance of Naive Bayes, linear Support Vector Machines (SVM) and k-Nearest Neighbour classifiers (k-NN). Naive Bayes performed best for estimating the five tasks that the authors had defined, while k-NN with $k = 1$ performed best in estimating labels defined by the participants themselves.

Devaurs, Rath, and Lindstaedt (2012) and Rath, Devaurs, and Lindstaedt (2010) also compare Naive Bayes and k-NN classifiers as well as J48 decision trees and linear SVM. In addition to features from the information objects in the data, keyboard strokes, mouse events and other interaction features are used in the classifier. These features are managed in their ontology-based user interaction context model, UICO. The best classification results were obtained with J48 decision tree and Naive Bayes classifiers.

A clustering method is described by Brdiczka (2010). Their task reconstruction system uses a spectral clustering algorithm to find task clusters based on the temporal switch history.

Furthermore, Armentano and Amandi (2012) used Variable Order Markov models with an exponential moving average to predict the user's goals from unix commands.

Koldijk et al. (2012) use a key logger to monitor a knowledge worker's activity with the purpose to track which tasks the user is performing. They investigate in a user study which task labels the knowledge workers intuitively use. These tasks include: read or write e-mail, write report, program, analyse data and search for information. Additionally, they investigated whether these tasks can be recognized automatically from the low level log events (such as mouse or key activity or the active application) using automated classifiers (SVM, Naive Bayes,etc.). They found that with relatively little data, i.e. a few hours, reasonable classification accuracy of 60–70% , depending on the user, could be obtained. However, there were many individual differences and there was no single classifier type that performed consistently over users.

In the SWELL project we use the work by Koldijk et al. (2012) to provide feedback to the user on the activity level, but for our identification of context we are more interested in a topic-based identification. The combination of feedback on activity level and on context/topic-level gives the best insight on how the user has spent his day, which is our goal in the 'user-context awareness' use case.

MEMORY-BASED CONTEXT RECOGNITION

Some authors interpret context recognition merely as a memory process, and only use temporal information to recognize contexts. They do not identify a context as label, but as a combination of tasks that were active at the same time: they memorize which windows were previously open together with the current window.

An example is the study by Abela, Staff, and Handschuh (2010), they propose a task-based user model that acts as a knowledge workers' mental model of a task, consisting of all computer resources related to that task. These should be used to resume a task-state after it has been suspended. The authors indicate the problem that different documents opened in the same application may belong to different tasks, complicating the method to be used for making a task snapshot.

Additionally, Omata, Ogasawara, and Imamiya (2010) propose a project-restarting system where files associated with a main file are automatically reopened. Associations between windows containing files and the importance of the window are automatically predicted. Features they use are window depth, visible representation ratio, and screen occupancy ratio.

Kersten and Murphy (2012) describe a task-focused desktop in which they present users with lists of documents associated with the tasks. The list is trimmed based on the frequency and recency with which a user interacts with the associated documents to determine whether it is still interesting for that task. They describe a longitudinal case study in which some university colleagues work with their system. The users manually start and stop tasks, during an active task all accessed documents are automatically associated with the task. The authors find that users tend to revisit tasks mostly the same day, suggesting that there is no need for auto-trimming.

These studies suggest that the memory-based approach is useful when we aim to support the user's work flow, but it is not usable to present the user with an overview of his day (use case 'user-context awareness'). Therefore, we focus on topic-based context identification instead.

6.3. CONTEXT RECOGNITION AND IDENTIFICATION USING AN INTERACTIVE ACTIVATION APPROACH

The conceptual model in Figure 5.2 described the elements that play a role in the context of a knowledge worker. In practice, the way the contextual elements influence each other is complex. To evaluate the model in a more straightforward task, we identify what a user is working on. The method that we present, however, is designed to be able to also take into account more complex tasks in the knowledge worker context, and a more diverse range of contextual elements than which we evaluate in this chapter.

For now, we describe a method to recognize and identify context. That is, we extract meaningful contextual information from the interactions with the computer (context recognition), and we attach a tag to it that the knowledge worker can interpret as one of the tasks he is working on (context identification). In the evaluation presented in Section 6.4 this task tag is the project name where the current activities belong to. We continue with a description of the contextual information that we use, after which we describe the novel context recognition and identification method.

6.3.1. CONTEXTUAL INTERACTIVE ACTIVATION MODEL (CIA)

In previous work (see Section 6.2.2), the information that has been used to recognize context can be categorized in the following dimensions: time, terms or topics, social information and location. In Figure 6.1 we visualize the literature that has been described in Section 6.2.2 in terms of the types of information that they have used.

In contrast to previous literature, the SWELL project aims to integrate all four dimensions of contextual information. Especially when working with multiple sources of data it is important to realize that different types of information sources have different characteristics that need to be dealt with appropriately. For example, in e-mail the sender and receiver are very important for categorizing the message, while for documents the content is more important (Sappelli, Verberne, and Kraaij, 2012; Sappelli, Verberne, and Kraaij, 2014). Considering that both email and documents are important parts of a knowledge worker's activities, it is important to be able to use both topics as well as entities such as person names as inputs for context recognition.

The difficulty in context identification is how to combine the various dimensions in an effective manner. In the method that we describe we have chosen a cognitively plausible approach that associates contextual elements to each other without the need to explicitly define the relations between them. The human brain is constantly making associations between observations (Anderson and Bower, 1973), and the intuition is that modelling these associations is the key to understanding how an individual would interpret his context. For example:

> The project 'SWELL' could be described by the terms 'stress', and 'knowledge worker' and the time period '2012'. If at some point in time the term 'burnout' is observed, we will most likely ascribe the logged activity to the project 'SWELL'. Although there is no direct association between 'SWELL' and 'burnout', we can find an indirect association be-

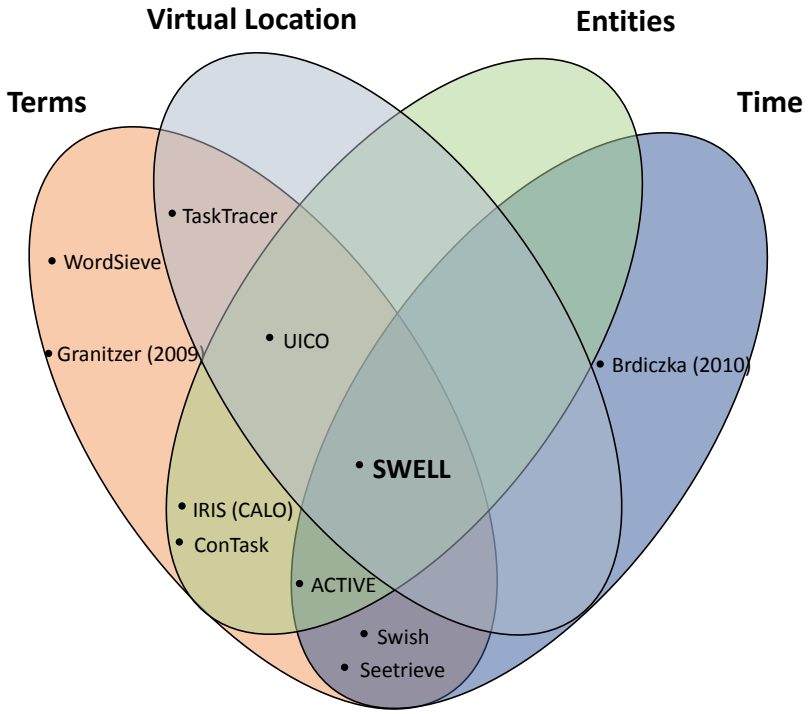


Figure 6.1: Visualization of literature and which information each project uses to link documents or events.

> tween 'stress' and 'burnout', leading us to a correct classification of the observed activities into the project 'SWELL'

Another motivation is that these associations could give insight in the behaviour of knowledge workers in terms of context switches. For example:

> A person is reading about 'Turing' in relation to the Turing test. But, through recent associations with the movie 'The imitation game', the knowledge worker is distracted, switches his context and reads up on the latest news about 'Benedict Cumberbatch'.

The intuition of the importance of associations has inspired us to adapt a well-known cognitive model for word recognition; the interactive activation and competition model (IA model) by McClelland and Rumelhart (1981). The IA model is a feed forward model that assumes that letter features stimulate relevant letters, letters stimulate relevant words and finally words stimulate relevant letters again. Within

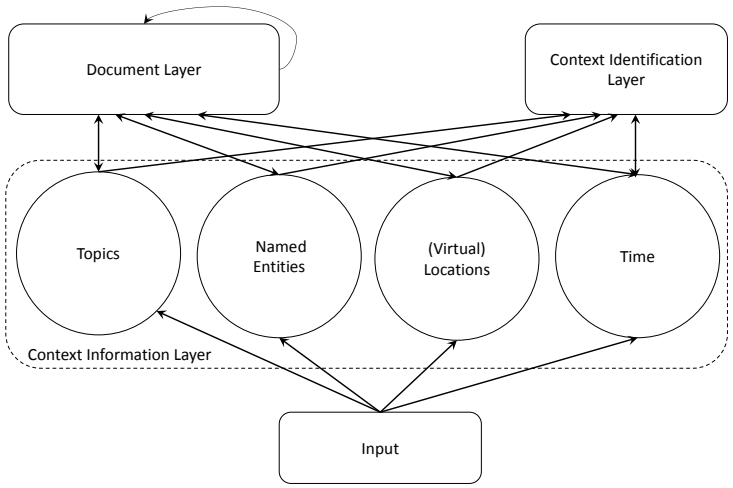


Figure 6.2: The Contextual IA model (CIA): a networked approach for context recognition and identification

each level there is competition; each feature inhibits other features, each letter inhibits the other letters etc. A rest activation and decay function complete the model. The model has been successfully applied as a cognitive model for bilingualism (Dijkstra, Van Heuven, and Grainger, 1998) as well.

CONSTRUCTION OF THE INITIAL NETWORK

We have adapted the IA model to use it for context recognition and identification (see Figure 6.2). Hereafter it will be referred to as the Contextual Interactive Activiation-model (CIA). It is constructed as follows: First, we define three layers in the model:

- the document layer: this layer contains all information objects that a user writes or reads and includes web-documents and emails.
- the context information layer: this layer contains the context information, divided into the four categories of context information types (terms or topics, entities, locations and date/time elements).
- the event layer: this layer is the input for the network. Here, recorded events from a key-logger, collected in event-blocks, enter and activate the network. In the case of computer activity one possible instantiation of an event-block can be a collection of events (key activity, mouse activity, window title, URL) that was recorded between opening a tab or window of an application and switching to another tab or application or closing it. This would mean that the event blocks are variable in their duration.

Table 6.1: Connection strengths between the various node types. These are the weights on the activation flow from one node to another. They are based on the well-known Information Retrieval concept tf-idf term weighting. Other choices for connection strengths are possible. $\#outlinks$ refers to the number of outgoing connections of a node.

From	To	Value or function	Motivation
Event-block	Date/Time	1.0	An event has one unique time stamp
	Entity	$\frac{\#entity_x \in event}{\#entities}$	Strength of activation of an entity should be dependent on how strong the entity is present in the event, proportional to the number of all entities in the event
	Location	1.0	An event has at most 1 location
	Topic	$\frac{topic_x \in event}{topic_{1..n}}$	Strength of activation of a topic should be dependent on how strong the topic is present in the event, proportional to the number of all topics in the event
Date/Time	Document	$\frac{1}{\#outlinks}$	Multiple documents can be accessed on the same date, or in the same hour.
Entity	Document	$\frac{1}{\#outlinks}$	IDF type measure; entities that occur in many documents should be less influential
Location	Document	1.0	
Topic	Document	$\frac{1}{\#outlinks}$	IDF type measure; topics that occur in many documents should be less influential
Document	Date/Time	1.0	
	Entity	$\frac{\#entity_x \in document}{\#entities}$	Strength of activation of an entity should be dependent on how strong the entity is present in the document, proportional to the number of all entities in the document
	Location	1.0	A document only has one location
	Topic	$\frac{topic_x \in document}{topic_{1..n}}$	Strength of activation of a topic should be dependent on how strong the topic is present in the document, proportional to the number of all topics in the document
Document	Document	$\frac{1}{\#outlinks}$	A document can have a relation to multiple other documents

Each of these layers contains nodes, and each node contains connections to nodes in another layer. Each node is a nominal version of the variable. Time nodes include nodes for each year, for each month in the year (January–December), for each day in the week (Monday–Sunday), for each day in the month (1-31), for each hour in the day (0-24) and for each quarter in the hour (0,15,30,45). Each of the

locations, entities, topics or terms has a single node. Since entity recognition and topic recognition can be probabilistic in nature depending on the method of choice, probabilistic properties can be enforced using the connections. For example, the probability that a topic is observed in an event determines the connection weight between that event and the topic. Similarly the probability that a topic is observed in a document determines the weight of the connection from the document to the topic.

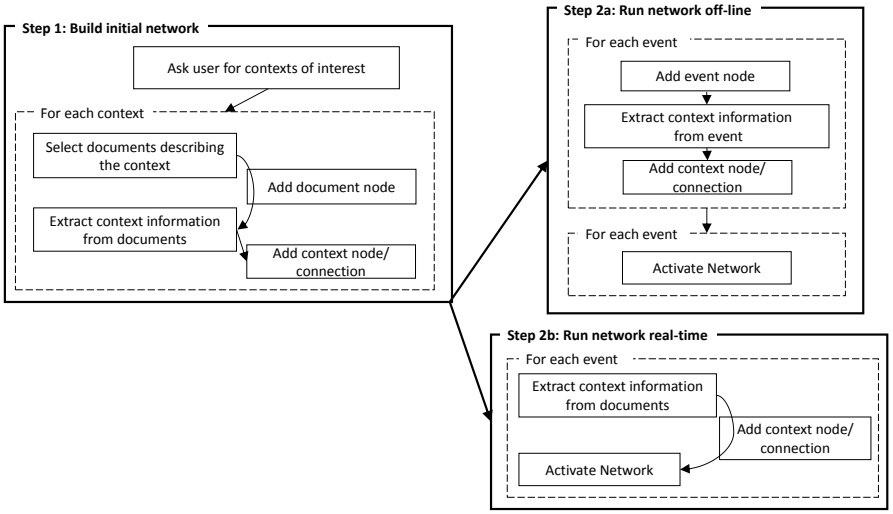


Figure 6.3: Work flow of building and running the CIA model

An initial network is built top-down (step 1 in Figure 6.3). The documents on the computer of the user are collected and for each of them a node is created on the document layer. For each document the location attribute is recorded, and the topics and entities are extracted using topic and entity recognizers. These context information elements are represented by nodes on the context-level. Bi-directional connections are made between each of the extracted context nodes and the corresponding document node. The strength of the connections of document to context differs from the strength of context to documents. An overview of the connection weights and the motivation for the various weights is given in Table 6.1. Presently we only use excitatory connections (stimulation connections) in our network and no inhibitory connections in order to limit the complexity of the model. Another reason is that in contrast to word recognition, multiple contexts can be validly active at the same time, so there is no need for competition between the contexts. There are no within-level connections at the context level, but there are on the document level, which will be clarified later on.

RUNNING THE NETWORK

Essentially we have an initial model now that describes the associations an individual may have made based on information that is already on their computer. In this

chapter we focus on context identification to provide the user with an overview of his day ('user context awareness'), a task which is executed only once a day (step 2a in Figure 6.3). For this purpose we can enhance the model bottom-up with each event block (coming from a key logger) that is observed. Incoming event blocks are temporarily added to the event layer. From these event blocks context information is extracted to create context nodes and connections, similar as in the documents, but now based on the event information. There is always a date/time-stamp for the event. If there is sufficient content information, originating from window titles, typed keys and caption information, entities or topics can also be identified. If there is a reference to a document in the event block, this document is added to the document layer. Connections are made from the date and time elements of the event block to the document. Since temporal proximity has been used successfully as a feature for context recognition (Warren et al., 2010; Štajner, Mladenić, and Grobelnik, 2010), connections are made from the document in the current event block to the document in the previous event block. This is a within-level connection between two documents based on their temporal proximity.

In contrast to adding the events to the event layer in the building process, we could also complete the network simultaneously while running it to provide real-time support (step 2b in Figure 6.3). This is necessary for the 'working in context' scenario where the user is supported during his work with context-aware functionality such as notification filtering or finding relevant documents.

Running the network entails that the event nodes (input nodes) are activated and that the activation is spread through the network using an activation function. The activation procedure has 3 steps:

- First the event block is activated in the event layer. The activation of the node is set to the maximum.

- Then the connected context nodes are activated using Grossberg's activation function, which runs for several iterations. The difference in activation from one iteration to the next is defined as follows:

$$\delta a = (max - a)e - (a - min)i - decay(a - rest) \tag{6.1}$$

  where a is the current activation of a node, e is the excitatory input of the node, i is the inhibitory input and min, max, $rest$ and $decay$ are general parameters in the model. The excitatory input moves the activation towards the maximum, while the inhibitory input moves it towards the minimum. The decay parameter causes the activation to return to its resting level when there is no evidence for the node and allows for cross-over of network activation from one event-block to the next.

- In the next iteration of the activation function these context nodes stimulate their connected nodes and this continues for several iterations such that all levels in the network get properly activated.

LEARNING ASSOCIATIONS BETWEEN CONTEXT AND TAG

The steps up to now allow us to activate the network, making it possible to *recognize* the active context in terms of an activation pattern over context information elements. We cannot, however, *identify* context yet. For that purpose we need some additional nodes; context identification nodes. These nodes represent the context identification tags described earlier. An example of suitable labels are project names, which we evaluate in Section 6.4.2.

In order to *identify* the active context, the model first needs to learn what the contextual elements are that are associated with a certain context identifier. These associations reveal which connections to make between context nodes and context identification nodes. The identification nodes have no outgoing connections. The activation level of an identification node signals which context was most likely active during the event block.

There are three possible approaches to determine which connections between context nodes and identification nodes need to be made and how strong they should be related. The first is a manual process where the user would be asked to describe each context identification tag with a couple of keywords or entities that are related to it. These terms are then the first context nodes in the network. A downside to this approach is that it is possible that the data, whether it be event blocks or documents, might not contain the exact descriptions of the user. The connected nodes might be very sparse. Additionally it is difficult to properly weigh the connections.

The second approach is a supervised one, where the network would be presented with a subset of event blocks that are labelled with their context identification tags. Connections can be made between the elements in the events (topic, time, location and entity) and the context identification tag with which the event block is labelled. This will result in more connections than in the manual approach, but the downside is that the event blocks from the key logger need to be labelled first.

The final approach is an alternative method and is related to question 2: "How can we implement the model for context identification in a way that requires as little labelled data as possible?". In this approach transfer learning is used. One method of transfer learning is a method where labelled items from a source domain are used to train a classifier in a (different) target domain(Arnold, Nallapati, and Cohen, 2007; Bahadori, Liu, and Zhang, 2011). In the knowledge worker case the system makes use of documents on the user's file system as source domain, to be able to classify event blocks; the target domain. For that purpose, each project folder name on a user's computer can be used as a context identification tag. The documents in that folder can serve as the training data for the connections that need to be created between the contextual elements and the context identification tags. Thus, contextual elements are extracted from the documents in the project folder and connections are made accordingly. With this method, no labelled data, other than a couple of organized documents, is needed and the connections can be weighted according to their strength of occurrence in the documents. This is a type of transfer learning where document categorization is used as a source for initializing a network for the purpose of the categorization of events into contexts (Arnold, Nallapati, and Cohen, 2007; Bahadori, Liu, and Zhang, 2011). In essence the approach is of the type feature-

representation transfer (Pan and Yang, 2010). The context layer in the network can be seen as a feature representation that represents both the source domain (documents) and the target domain (events) and reduces the difference between the two. We have applied and evaluated this method successfully in the domain of e-mail categorization (Sappelli, Verberne, and Kraaij, 2014). In the remainder of this chapter we focus on using the network in a transfer learning setting.

For clarity we want to add that the learning aspect in our network, albeit unsupervised, supervised or by transfer learning, only entails learning which connections should be made between context level and context identification level, and not which weights are optimal.

6.4. IMPLEMENTATION AND EVALUATION

In this section we will evaluate how well we can identify the context of the user using the proposed model. For the proposed 'user-context awareness' use case in Section 6.1 (giving feedback on how the user spent his time) it would suffice to evaluate classification power on hourly time frames. However, for the 'working in context' use case, where information needs to be filtered directly, context identification on small time frames may be needed, even though in that case the information in the context layer may also be used directly rather than to make an identification step first. Since the focus of this chapter is on context identification and it is easier to go from small time frames to larger time frames, we evaluate the output of the network on a per event-block basis (Sappelli et al., 2014), where the average duration of an event block is in the range of seconds.

6.4.1. DATA

For the evaluation we use a dataset of event-blocks originating from human-computer interaction data representative for a knowledge worker's activities which is labelled according to the activity that was carried out during the event block (Koldijk et al. (2014) and Sappelli et al. (2014)). This dataset is publicly available. The blocks were collected during a controlled lab-experiment where 25 students were writing reports and preparing presentations on 6 subjects (e.g. a road trip in the USA, Napoleon) while they were being monitored by various sensors. During the experiment two additional subjects were introduced, resulting in a total of 8 context identification tags, that we aim to recognize. In a real office settings these tasks would be the various projects that a user is working on. We only use the sensor-data from the installed key logger (uLog v3.2.5) and the file history (coming from IEhistory).

Labelling The data was labelled using Amazon Mechanical Turk. The annotators were presented with a mimicked version of the original desktop view of the event. Additionally a field with the typed keys was presented. The annotators were asked to choose one of 8 tags, corresponding to the subjects, and an additional 'unidentifiable' tag. They were also asked how certain they were of their answer on a 5 point scale, with 5 being completely certain and 1 being not certain at all.

The dataset consists of 9416 labelled event blocks, with an average of 377 event blocks per participant. The distribution of the labels, excluding unidentifiable labels, is quite skewed as can be seen in Figure 6.4. The labels 'Einstein' and 'Information Overload' have less event blocks, since these were not main tasks. The labels 'Perth' and 'Roadtrip' occur relatively often, most likely because these tasks required more searching and planning, and with that a higher variety in sources.

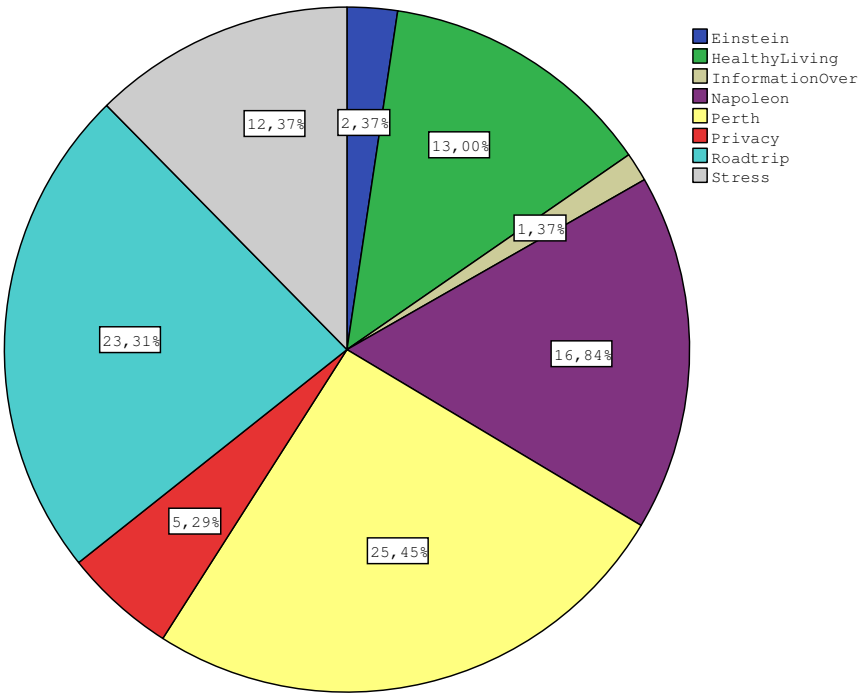


Figure 6.4: The distribution of the identifiable labels in the data for context identification.

6.4.2. IMPLEMENTATION DETAILS

For the extraction of topics from the documents and event blocks we use a latent dirichlet allocation model (LDA model), which is often used for topic extraction. In this setting we have used the MALLET implementation of LDA (McCallum (2002)) and 50 topics are extracted. The initial LDA model is trained for 1500 cycles on a set of manually selected Wikipedia pages (e.g. the Wikipedia page 'Napoleon' for the topic Napoleon), one for each of the tasks from the experiment. In a real office setting, these documents could be project description documents. Document inference (i.e. determining the topics for a new unseen document) is also based on sampling for 1500 cycles. The input for inference on an event consists of text from window titles, typed keys and captions.

For the entity extraction, the Stanford entity extractor trained on English is used.

This trained model provides good results for the entity classes Person, Organization and Location, which are the ones we use in the network (Finkel, Grenager, and Manning (2005)). Again inference is done on either document content or event content (text from window titles, typed keys and captions)

The date and time nodes on the context layer in the network consist of separate nodes for day of the week, day of the month, month, year, hour and minutes rounded to 00, 15, 30 and 45. The location nodes in the network are the file folder in the case of files on the computer and the domain name in case of web-documents. In this dataset there is no access to e-mail messages related to the topics.

Normally the network would be run until the activation converges. For efficiency we run each event block for 10 epochs (iterations), which is sufficient to activate all levels in the network, and keeps the running time low. Efficiency is important for our application scenarios ('working in context' and 'user context awareness') as well, but in some settings we may want to distinguish between short – little impact – event blocks and longer event blocks, which are probably more important. In that case we could run each block for a number of epochs corresponding to the duration of the event block times 10. This means that an event-block with duration 1 second, is activated for 10 epochs, so that there is enough activation in each of the levels, but an event-block with a longer duration is run for 100 epochs and thereby has much more impact on the overall activation in the network.

The identification of a context is based on the node with the highest increase in activation for an event block, compared to the node's resting level of activation. This is necessary because the network does not necessarily converge within 10 epochs. By looking at the increase in activation rather than the highest absolute value of activation, the network focuses on the evidence in the event block. This ensures that nodes that are decaying are not preferred over nodes that have an increase in activity, even though the absolute activity of the decaying node may be higher than that of the increasing node. By using the increase relative to the node's own resting level we prevent that nodes which activity levels increase slightly because they are already highly activated are unjustly ignored. The resting level of nodes may vary due to the number and strength of incoming connections. Nodes that have many incoming connections are more likely to always receive a little bit of activation, preventing them from returning to the global resting level.

These settings have been tweaked based on the data of participant 2, whose data has also been used to optimize the parameters of the various algorithms (See Section 6.4.2.1).

PARAMETER OPTIMIZATION

There are six parameters in the original IA model. Additionally, the LDA model we use has some additional parameters: the number of topics and the number of iterations. We used data of one of the participants (person 2) as development data, and used classification accuracy as our optimization measure. We first optimized the LDA parameters using the default IA parameters, resulting in an LDA setting with 50 topics and 1500 iterations. Then, using those LDA settings, we optimized the IA parameters with a hill-climbing approach starting from the default parameters. We found no set

Table 6.2: Parameter settings used during evaluation

Parameter	Definition	Value
α	Strength of excitation	0.1
γ	Strength of inhibition	0.1
Min	Minimal value of activation	−0.2
Max	Maximal value of activation	1.0
$Rest$	Resting-level of activation	−0.1
$Decay$	Strength of decay	0.1

of settings that was significantly better than the default, so our final parameter set is the same as the default parameters from IA as presented in Table 6.2.

6.4.3. UNDERSTANDING THE CIA APPROACH

Before we dive into the performance of the network and compare it to baselines in the next section, we first want to show what is happening in the network at run time, and why we think this is useful for the problem at hand.

The main issue in our context classification problem is that the data that we can observe, namely the event blocks with key-logging information, is very sparse and noisy. Since we focus on window titles and typed keys there is not much data that can be used. A window title only contains a couple of words, and these are not necessarily related to the content of the window. Typed keys may include more words, that however could be expressed in the wrong language, or contain typing errors and corrections, resulting in incomplete or erroneous data. An example event block is presented in Table 6.3. The network approach allows the expansion of the sparse observed data with information that is associated with the observed input. For example, a single recognized entity such as 'Los Angeles' in event block 59 is likely to occur in some document about the USA. By activating the USA related documents, other entities such as 'united states' and 'barack obama' are activated, and activation of USA-related topics is enhanced. This increases the likelihood that the correct label USA is assigned.

Not only is the data that we observe sparse, the data is represented by different types of features such as topics, entities, location and time information. It is important that the number of features for a type does not influence the result. For example, in the LDA model we have 50 features, while for the entity model we might have many more. The entity features should not outweigh the topic features, simply because there are more of them. In the network, features that are not activated have little impact on the overall activation. Thus during activation it does not matter that there are more entity features than topic features, since most entity features will not be observed and not activated.

An important aspect in activating the network is the decay parameter. This parameter ensures that past information is not immediately forgotten. For context classification this is useful, because typically there is a dependency between one event block and the next, when the knowledge worker is working on the same task. The

Table 6.3: Example Event Block

id	42
time	20120919T133339573
app	WINWORD
window title	
typed keys	Australia is a cp ountry that a country knkw with a number of a c0 country know n for its con untless natural wonders va and a one among thr mos e amoin ng the top tpurist destinatioop n ns that the tourust visiti. s wa o scemo nic beauty. It Austrai l West Australia o Perh the th isth ca Australia consists of many beaitif utiful cities
domain	
caption	Close Document1 - Microsoft Word Document1 - Microsoft Word

effect of decay is especially clear when the network is run for few iterations. The iterations help to smooth the boundaries between event blocks (Figure 6.5), because history-information is taken into account. When the network is run until convergence the boundaries between event blocks are more clear (Figure 6.6) because in that case the focus is on the evidence from the current event block instead of the history. Both figures show that the Healthy Living and Stress labels are in competition with each other.

6.4.4. RESULTS

In this section we analyse the model in terms of its classification performance of the event blocks. We start with a comparison of the proposed network, CIA, to existing approaches k-NN and Naive Bayes. Then we analyse the effect on performance of each of the context information types. We continue with a comparison of the network to one that is built on the fly and that can do real-time context identification. We end with the analysis of the influence of language on the performance of the model, since non-English knowledge workers often use a mixture of languages.

ACCURACY OF THE CIA-MODEL

In Table 6.4 we present the accuracy of our CIA model with LDA topic recognition (a). These results are the average over the 25 participants over 25 runs per participant. The need to compare multiple runs per participant stems from the random nature of the LDA model (We will elaborate on this in Section 6.5). We cannot conclude from one run that this is a representative outcome of the model, so we average over multiple runs.

In addition we provide results for a CIA model where the topics are based on term extraction rather than the LDA topic model (b). In this setting the topic model consists of uni-grams, bi-grams and tri-grams that are extracted using the method

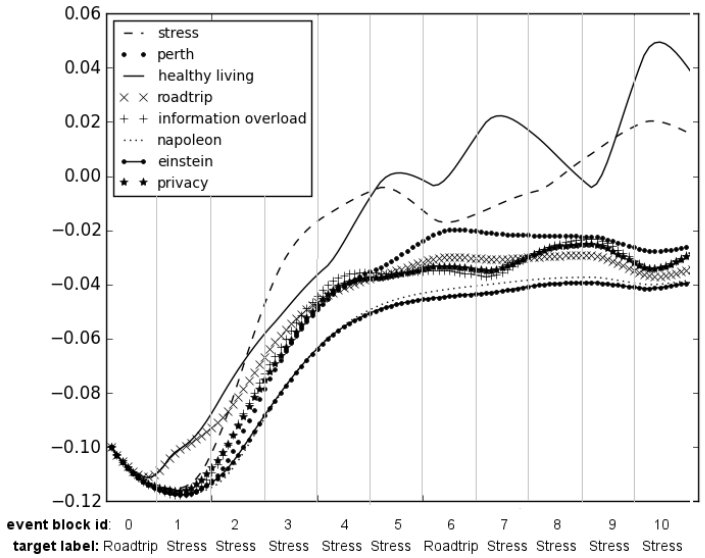


Figure 6.5: Activation on the identification level with 10 iterations per event-block. The x-axis shows the id-number and the target label of the event block.

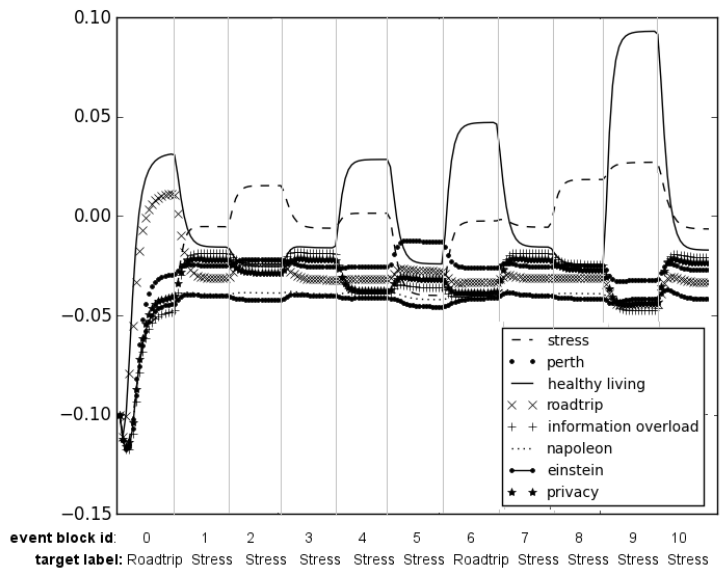


Figure 6.6: Activation on the identification level with 100 iterations per event-block. The x-axis shows the id-number and the target label of the event block.

Table 6.4: Comparison of the CIA model to k-NN and Naive Bayes baselines in the identification of context

Method	Avg Accuracy
a. Context IA using LDA	**64.85%**
b. Context IA using Term Extraction	61.56%
c. k-NN with $k = 1$	55.83%
d. k-NN using LDA with $k = 1$	59.75%
e. Naive Bayes	48.32%
f. Naive Bayes using LDA	60.49%
g. Majority Baseline	25.30%

described in Verberne, Sappelli, and Kraaij (2013). For each of the document folders (the projects of interest) that consist of 1 Wikipedia document each, we extract the top 1000 terms after which stop-word uni-grams are removed. The importance of a term is determined by comparing its frequency against its frequency in a corpus of general English (Corpus of Contemporary American English), after which hapaxes and partial terms (i.e. uni-grams or bi-grams that are part of another bi-gram or tri-gram in the list) are removed. In effect, each project category contains 548 salient terms on average . Examples of extracted terms are "world health organization", "post-traumatic stress", "American citizen", and "dementia". The 8 lists of salient terms are pooled together, resulting in a list of 4206 terms. These are used as nodes in the network, with the connection strengths as motivated in Table 6.1

The remaining results in Table 6.4 are obtained using Weka and are used as baselines to compare CIA to. Results c. and d. originate from a k-NN classifier, with $k = 1$ (optimal k) and e. and f. originate from a Naive Bayes classifier. The results for these methods have been obtained using 10 fold cross-validation on the event-block data. This means that 90% of the event-block data was used for training and 10% for evaluations, so the type of training data is different from the training data of the CIA model. The CIA model does not need examples of labelled event block data. Baselines c. and e. receive the same raw input as the CIA model receives during run-time (e.g. window title, typed keys, caption, URL), but without the additional pre-processing that CIA uses for topic and entity determination. The k-NN and Naive Bayes classifiers are, however, provided with vector representations of the full content of the document if there was a reference to a document in the event block as additional features. This is common for current approaches to context identification (Cheyer, Park, and Giuli, 2005; Stumpf et al., 2005; Granitzer et al., 2009; Devaurs, Rath, and Lindstaedt, 2012). Baselines d. and f. were obtained by additional feature extraction using LDA topic recognition and entity extraction using the Stanford entity recognizer. This results in the same feature set that is used in the CIA model on the context level.

Table 6.4 shows that the CIA network with LDA (a) has an increased performance over both k-NN (c,d) and Naive Bayes (e,f). The difference is significant in a 2-tailed t-test with $P < 0.001$ regardless of whether the improved feature extraction was used or not. The feature extraction using LDA and entity extraction does improve the quality

of k-NN and Naive Bayes classifiers. The contextual IA model with term extraction (b) is significantly better than Naive Bayes (e), $P < 0.001$, but not better than k-NN (c,d) or Naive Bayes with LDA modelling (f), $P >= 0.381$. There is no significant difference between the CIA model with LDA and with term extraction.

Between the runs with the LDA model (a) there was an average spread of 10.46 percent point in accuracy over participants (average minimum 59.37%, average maximum 69.84%). This means that depending on the specific LDA model that is used, there can be a large difference in performance of the model, even when it is trained on the same data each initialization. This variation also occurs in the k-NN and Naive Bayes runs with LDA as feature selector (d and f). A possible explanation could be that this variation in performance is a result of the size of our corpus (only 8 documents), however results with larger corpora (either 4582 documents – all websites that were observed during the entire experiment – or 6561 documents – the 8 Wikipedia including their outlinks respectively) showed just as much performance variation. A repeated measures analysis where each run is seen as a measurement for a participant showed that the variation between runs was not significant ($P = 0.344$ in CIA).

When we look at the precision and recall values for the various classes, it is clear that some classes are more easily recognized than others. Perth and Roadtrip have high precision regardless of the classification approach, while Einstein and Information Overload have low precision. One explanation for this finding is the type of assignment that Einstein and Information Overload originate from. Both these topics were short questions asked via e-mail during the experiment to distract the user rather than the assignment to write a report or prepare a presentation. Because these assignments were smaller tasks, they occur much less often in the data. Napoleon has a remarkably high precision in CIA, compared to k-NN and Naive Bayes. Overall CIA seems to have a little higher recall compared to k-NN and Naive Bayes. k-NN tends to have a higher precision than recall.

EFFECT OF PERSONAL WORKING STYLE

The CIA approach (using LDA or term extraction) gave the best accuracy in context identification for 84% of the participants. For the remaining participants, Naive Bayes using LDA was the best approach. This shows that CIA is a robust approach that is not influenced much by personal working style, which is in contrast to the findings by Koldijk et al. (2012)

When we analyse the results of one of the participants for which both CIA using LDA performs well (participant 8, average accuracy 78.54%) and for which CIA performs poorly (participant 4, average accuracy 44.60%) there seem to be a few characteristics of the data that may have played a role. First of all, the participant 4 has fewer event blocks than participant 8 (123 compared to 475), meaning that the duration of the event blocks was longer. However, we have found no significant Pearson correlation between the number of event blocks and the accuracy when we take all participants into account.

Second, participant 8 seems to be a copy-cat; he pasted text (copied from a web page to a document for example) in 20% of his event blocks, while participant 4 only

Table 6.5: Precision, Recall and F1-measure for CIA with LDA and the best k-NN and Naive Bayes baseline runs.

Run	Precision			Recall			F1-score		
	CIA	k-NN	NB	CIA	k-NN	NB	CIA	k-NN	NB
Einstein	0.32	0.36	0.12	0.44	0.27	0.32	0.37	0.30	0.17
Privacy	0.57	0.66	0.55	0.73	0.58	0.62	0.64	0.61	0.60
Information Overload	0.07	0.27	0.06	0.47	0.28	0.34	0.12	0.27	0.10
Roadtrip	0.70	0.69	0.63	0.49	0.60	0.60	0.57	0.64	0.61
Healthy Living	0.66	0.60	0.50	0.62	0.54	0.57	0.64	0.57	0.53
Perth	0.80	0.73	0.72	0.78	0.73	0.69	0.79	0.74	0.71
Stress	0.67	0.66	0.59	0.71	0.63	0.61	0.69	0.64	0.60
Napoleon	0.87	0.67	0.67	0.76	0.59	0.61	0.81	0.63	0.63
Average	0.58	0.58	0.48	0.63	0.53	0.55	0.58	0.55	0.49

did this for 5% of his event blocks. Since copied text is captured in the caption-data of the event-blocks this may have given the network a richer input compared to typed text. The typed text contains all keystrokes, including all typos and backspaces in case of corrections. It does not contain the resulting correct word, since it is not corrected for typing or spelling errors, which makes it a very noisy data source. Finally, the human annotators rated the confidence in their subject labels for participant 4 on average 3.97 on a 5 point scale with 5 being completely certain, while the annotators rated their confidence in labels for participant 8 on average 4.75. This suggests that the data for participant 4 might have been more ambiguous or unclear in general.

Another explanation is found in the majority class of the data for the participants. For participant 4, the majority class is Healthy Living, which comprises 38% of his data. For participant 8 the majority class is Perth which is 37% of his data. Since the precision for the class Perth is in general higher (0.8) than that of Healthy Living (0.65), it is likely that participant 8 will have more correct predictions that participant 4.

INFLUENCE OF LANGUAGE

Two of the participants (participant 1 and 7) wrote their reports in Dutch, so we expected that their performance would increase when we trained the LDA model on Dutch equivalents of the English Wikipedia pages. Unfortunately we did not have a Dutch model for the named entity recognizer. However, because of the similarities between Dutch and English, some of the important entities (Perth, Napoleon) could still be found. Figure 6.7 shows the results.

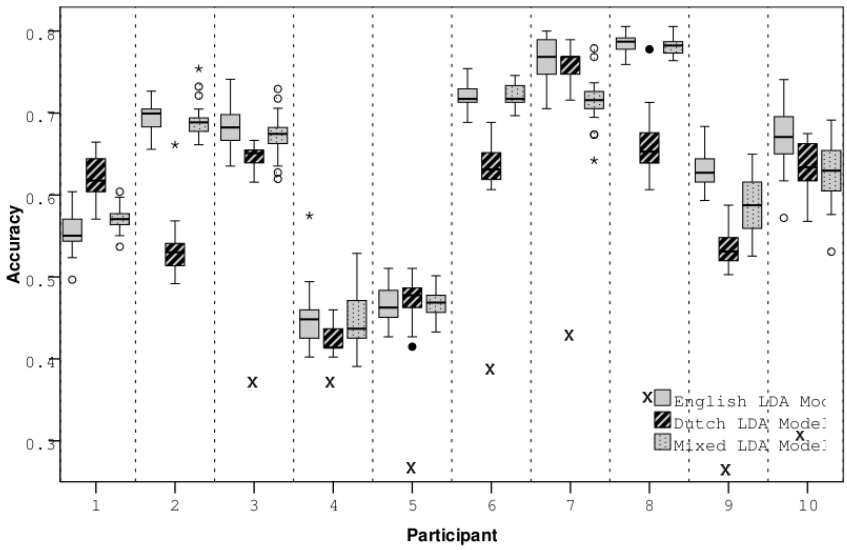


Figure 6.7: Influence of language on which the LDA is trained for participants 1-10. The majority baseline for the participant is represented with X. For participants 1 and 2 the majority baseline is < 0.26 and hence not in the figure

First of all it is interesting to see that the participants writing in Dutch do not stand out in terms of accuracy compared to the other participants. Surprisingly, the use of a Dutch LDA model lead did not have much effect on the accuracy for participant 7 (76.04% for Dutch compared to 76.59% for English). For participant 1 it lead as expected to an increase in accuracy (62.15% for Dutch compared to 55.60% for English). In total there were 4 participants that benefited from the Dutch model. There are several possible explanations why 3 of these participants benefited from the Dutch model even though they did not write their reports in English. The first explanation is that the documents in the Dutch corpus are not word by word translation of their English equivalents, so the actual information in the Dutch corpus might be different from the information in the English corpus. In general the Dutch corpus is a bit more sparse than the English, because the documents are shorter. It may well be that the English corpus contains more irrelevant topics. Another possible explanation can be that users, even though they wrote their reports in English, visited Dutch web-pages or issued their queries in Dutch. This may have been a side-effect of the fact that the homepage of the browser during the experiment was http://www.google.nl rather than http://www.google.com

Since the participants might have used a mix of both Dutch and English (the 'Mixed LDA model' in Figure 6.7), we used an LDA trained on both corpora as well. In general this approach performed worse than using an English model, but slightly better than using the Dutch model. This is most likely because the model finds separate Dutch and English topics, but still has a maximum of 50 topics, so compared to the models for 1 language, this model will have less fine grained topics. The mixed approach seems to have a preference for English topics.

INFLUENCE OF INFORMATION TYPE

One of our research questions was RQ2: 'What information is required for successful classification?'. For that reason we have run the network in several variants where we leave out either location-nodes, time-nodes, entity-nodes or topic-nodes. This provides insight in which elements are necessary for context identification and which are not so important.

The average performance of the various combinations of node-types is presented in figure 6.8. We see that a network with only entity nodes yields an average accuracy of 40.49%. However, only considering topic nodes, the network already has an average accuracy of 62.91% (significant improvement over just entities; $P < 0.001$). There is a slight but significant improvement ($P = 0.013$) to an average accuracy of 64.85% when adding the location, entity and time nodes compared to just topic nodes. The influence of time in the network is minimal.

Figure 6.9 shows the effect of using the term extraction instead of the LDA model. Again there are significant differences between the various information type combination, but now we see that a network with only term extraction gives an average accuracy of 58.96%. Again, the full network (terms, entities, locations and time) outperforms a network with just entities, or just topics ($P < 0.001$).

In both cases we see that the performance of the network is largely determined by the topic features. However, adding other information types can increase performance. Overall, time and location have little impact on the results. This is caused by

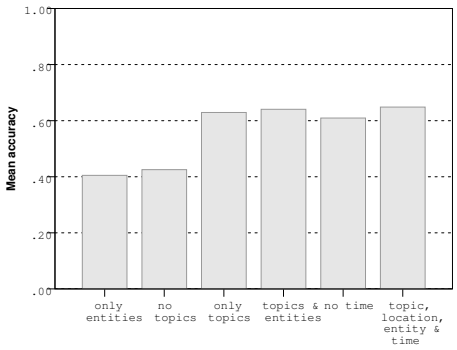


Figure 6.8: Influence of information types on classification performance of the CIA network with LDA

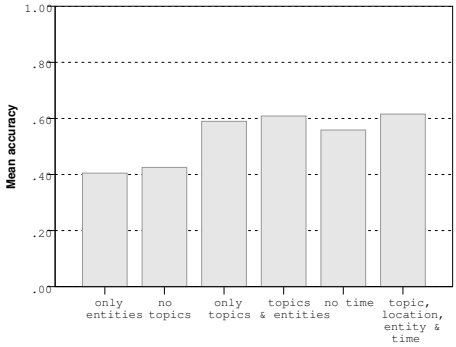


Figure 6.9: Influence of information types on classification performance of the CIA network with term extraction

the limitations of the data. We cannot estimate the influence of the time nodes realistically since each participant executed all their tasks on one day, so it will have little or no impact. Furthermore, our project-directories for training are only mock-ups and do not occur in the event-stream data, so no identification nodes can be activated based on location only, even though location is most likely strong indicator of a certain context in a realistic scenario.

CONTEXTUAL IA MODEL ON THE FLY

In the previous sections we initialized the model with all the event data after which we have run the network to determine the accuracy in context detection. This is consistent with our application idea where we present the user with an overview at the end of his day of how he spent his time that day.

We can, however, imagine a scenario in which we want to inform the user about his detected context immediately (for example for recommending relevant documents) as would be the case in our 'working in context' use case. Real-time context detection could be used to categorize information objects at creation time, or to get feedback from the user about detection accuracy which can be used to improve the network, or to filter out incoming information that is irrelevant for the current context and therefore might distract the user.

In this section we show the performance of the model when run on the fly. The initial training phase then only consists of training the LDA model and making connections between identification nodes and the entities and topics that are relevant for the identification nodes. All other nodes and connections such as visited websites will be created on the fly (i.e. real-time). Note, however, that no additional connections to the context identification nodes are made, so even though the events are labelled, we do not use this information. The on-the-fly addition of the events to the network, only increases the number of associations that are made between context nodes and document nodes, and document to document connections. The real-time accuracy (presented stream-data of 0%) is presented in Figure 6.10. The

figure also shows the delayed real-time performance (presented stream-data bigger than 0%). Thus, the accuracy of context prediction when the network has seen part of the event data already. For example, the system would start real-time prediction only after it has seen 10% of the data (about 18 minutes of data) compared to starting real-time prediction immediately (i.e. cold start). When the knowledge worker has been using CIA for a longer period of time, this can be seen as a delayed start of real-time prediction

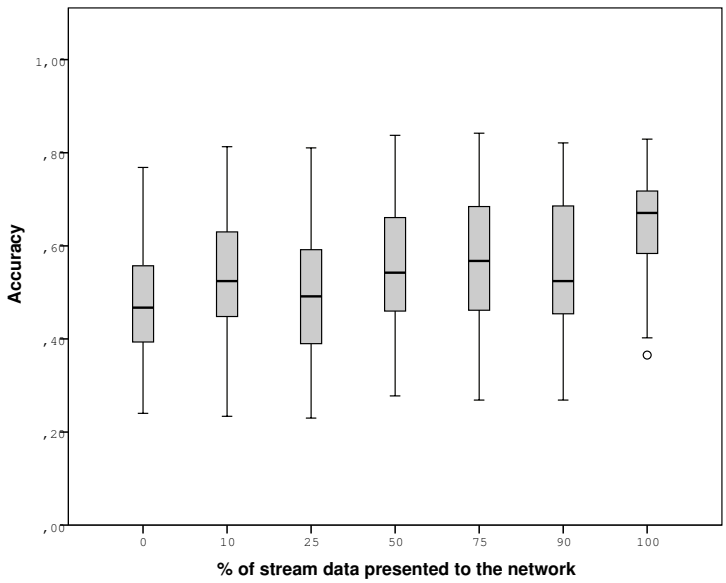


Figure 6.10: Classification performance of the CIA model with partial information presented to the model (0% represents real-time performance).

The results show that as expected the real-time performance is significantly lower than when the model is built based on all event data ($P < 0.001$). This is because the model starts with hardly any associations in its network. With few associations, the expansion behaviour of the network using the activation function is ineffective. Real-time performance (47.09% averaged accuracy) is equal to the non real-time Naive Bayes baseline ($P = 1.000$) but worse than non real-time k-NN ($P = 0.001$). These baselines are an unfair comparison, however, as they are *not* real-time and moreover trained on 90% of the event-blocks data, whereas CIA uses no labelled event-blocks at all. k-NN and Naive Bayes would have 0% accuracy in a real-time setting as they require labelled event-block examples beforehand. Reducing the number of training examples for k-NN and Naive Bayes reduces their performance, while running the network with a delay (i.e. presenting some data to it before running) increases the network's performance. Thus overall, the network has a clear advantage in a real-time setting or when little data is available beforehand.

6.5. DISCUSSION

In the previous chapter we presented a conceptual model for the knowledge worker's context. The model requires the notion of a knowledge worker, resources and a possible interaction between them. Primarily the knowledge worker is engaged with a task (the knowledge worker context) and the user can be distracted or helped by interaction with resources.

The implementation of a partial version of this conceptual model, CIA, has two main advantages. The first is that it needs very little labelled data to be used for context recognition and identification (RQ1). Instead of labelled event blocks, it uses one representative document for each context that needs to be identified. In the experiment we used one document per context identification tag, which was already sufficient for the task of context identification. This equals 8 labelled documents in total for the 8 context, compared to the 8474 event blocks (90% of all available event blocks) that need to be labelled for the k-NN and Naive Bayes baselines, which is a reduction of 99% in labelling effort.

The addition of more relevant documents might improve the confidence of the connections that are being made, although this is likely to be dependent on the quality and topics in the documents. Furthermore it could be beneficial to look into the usage of documents written or saved by the user as training examples. These provide insight in the focus of the user, which may help to personalize the context identification further.

A second advantage of the model is that it can expand sparse input data coming from a key logger to something more meaningful in terms of 4 types of information; locations, topics, entities and time. These are the types of information required for successful classification (RQ2). The model can make relations between the information types using the document layer. Because the model does not map input to identification directly, it can use the information on the intermediate levels as well. This makes it possible to support the knowledge worker in several ways using a single model for his context. An example is by context-aware document recommendation which we will validate in future work.

CIA is effective in classifying the user's context and has an average accuracy of 64.85% (RQ3).The main disadvantage of the model is that, when used with an LDA model for topic extraction, the accuracy shows a lot of variation due to the non-deterministic nature of the LDA output. The improved k-NN and Naive Bayes baselines with LDA extraction suffered from the same disadvantage.

6.5.1. LIMITATIONS

There are some limitations in our method of evaluation. First of all, even though time and location did not contribute to the performance of the model, we can not conclude that time and location are not important for context. This is because we expect that time and location become important when data includes multiple days and repetition of activities. In our data, which is collected during three hours on a single day, location and time will have little to no impact. The results of the real-time and delayed real-time performance also shows that more data in the network is better. This may indicate that when continuing the network the next day, the perfor-

mance could increase even more. Especially when you consider that the presented results are only of 3 hours of data. In future work we would like to investigate this by collecting data for multiple days of work.

Secondly, we evaluate the model on context identification, since this allows us to compare it to existing literature. However, this does not show the true purpose of the model, which lies in the possibility to use the context information layer itself. For example by using the activated context nodes as method to search or filter information. In future work we plan to address this, as soon as we can collect a suitable dataset.

6.5.2. FUTURE WORK

For future work, there are still many characteristics of the model to explore. First, we could use graph clustering techniques to see whether we can make the connections to identification nodes without user input.

Second, a disadvantage of the model is that it can become very complex when it has monitored a user for a while, because the number of nodes and connections in the network increases. Therefore we need to investigate possibilities to optimize and clean the model regularly to make sure that it runs efficiently. One possibility is to clean up obsolete connections and nodes; elements that have little added value in the network.

Furthermore, when labelled event-block data is available, we could optimize the weights in the network in order to improve the classification accuracy.

Additionally we would like to get a more complete overview of the knowledge worker's day. This means that we would like to incorporate diary information, or GPS-sensor information. This would help to recognize that the user is in a meeting or that he is travelling. One idea is to create sub-networks for different types of situations (physical activity, computer interactions, emotions, planned activities) and combine them in a larger network.

6.6. CONCLUSION

In the previous chapter we presented a new conceptual model for the context of a knowledge worker. The conceptual model was operationalized by an applied, cognitively plausible approach to context recognition and identification, which is the main contribution of this chapter. This applied approach was presented as the contextual IA model (CIA-model), which is adapted from the Interactive Activation model by McClelland and Rumelhart (1981). The model was evaluated on human computer interaction data that is representative of knowledge worker's activities. In the task of context identification the model performs at least as well as, but in general better than, k-NN and Naive Bayes baselines. Since the evaluation dataset is publicly available (Sappelli et al., 2014), the current work can be used as a new baseline for context identification.

We summarize the advantages of CIA as follows:

1. CIA is at least as good, but in general more successful in identifying the active context than k-NN and Naive Bayes baselines

2. CIA tremendously reduces the required labelling effort in comparison with k-NN and Naive Bayes by using a form of transfer learning.

3. CIA can deal with sparse and noisy inputs by making use of associations in the network to expand the input to more meaningful elements.

4. CIA is flexible in the type of information that is represented in the context information layer, creating opportunities for many application areas.

5. CIA is robust against differences in personal working style

The main disadvantage of CIA, is that the method for topic extraction used has a large influence on the overall performance of the model.

In the introduction we identified two use cases for which we can use the CIA model. For the proposed use case 'user-context awareness', where we provide the user with an overview of how he spent his day, or how long he spent on each activity, an average of 64.85% accuracy may not be sufficiently accurate, even though it's better than the alternative method. However, especially in the case of hour tracking in a company setting it would suffice to find estimates of longer time periods (e.g. hours). This would be an easier task than to find the correct label for each sparse event block of a few seconds separately.

For the 'working in context' use case, the identification needs to be more precise. This means that the accuracy needs to be improved. For this use case though, the context information layer and document layer could be used directly to filter information without the need of *identifying* the context first. We will explore this in ongoing research.

In future work we will evaluate the CIA-model on a real-time context-aware information delivery task (contextual support). In addition we will work on methods for automated context discovery. This would make the model more flexible, as it would create the possibility to remove and add new contexts to be identified on the fly.

III

Context-aware Support

Having an intelligent secretary does not get rid of the need to read, write, and draw, etc. In a well functioning world, tools and agents are complementary.

Alan Kay

7

COMBINING TEXTUAL AND NON-TEXTUAL FEATURES FOR E-MAIL IMPORTANCE ESTIMATION

Edited from: **Maya Sappelli, Suzan Verberne, Wessel Kraaij** (2013) *Combining textual and non-textual features for e-mail importance estimation*, Proceedings of the 25th Benelux Conference on Artificial Intelligence (BNAIC 2013).

In this work, we present a binary classification problem in which we aim to identify those email messages that the receiver will reply to. The future goal is to develop a tool that informs a knowledge worker which emails are likely to need a reply. The Enron corpus was used to extract training examples. We analysed the word n-grams that characterize the messages that the receiver replies to. Additionally, we compare a Naive Bayes classifier to a decision tree classifier in the task of distinguishing replied from non-replied e-mails. We found that textual features are well-suited for obtaining high accuracy. However, there are interesting differences between recall and precision for the various feature selections.

7.1. INTRODUCTION

In the COMMIT project SWELL (smart reasoning for well-being at work and at home[1]) we aim to support knowledge workers in their daily life. In the at work scenario one of the objectives is to prevent negative stress, either by coaching the user on his work style or by signalling stressed behaviour (Koldijk, Neerincx, and Kraaij, 2012). Another objective is to filter irrelevant information to preserve the user's work flow. A large source of incoming information for knowledge workers is e-mail.

[1] `www.swell-project.net`

For this latter objective there are three things that are important. First we need to know what the user is doing to determine which incoming messages are relevant for his current work, and whether presenting the user with the message is disturbing. We define this as recognizing the user's context. Second, we need to decide which incoming messages are important enough to present it to the user regardless of what he is doing. This aspect is important to make the user feel in control, i.e. that he does not feel like he is missing information. Third, it is important to understand why an incoming message is important or relevant so this can be used as feedback to the user (i.e. transparency).

In this chapter we aim to predict whether or not a receiver will reply to a message. We believe that, although the likeliness of reply is not the only factor determining message importance, replying to a message is a good indicator that a user finds this message important, otherwise he would have ignored it. This work is meant as a first step towards developing an e-mail client that helps to protect the user's work flow. Existing literature on the topic of reply prediction (section 7.2) focuses on features such as the number of question marks and the number of receivers. We aim to investigate the influence of the textual content of the message on the likeliness that a receiver will reply to the message. This can also be used to make it transparent to the user why a classifier beliefs that the user needs to reply to a message. To this end, we train classifiers with various feature sets and compare their results.

7.2. RELATED WORK

This section presents an overview of the literature related to reply prediction. First, we present some general work on email responsiveness. After that we present some previous attempts to manual or automatic prediction of whether an e-mail message is going to be replied to.

Tyler and Tang (2003) conducted a study to email responsiveness to understand what information is conveyed by the timing of email responses. They used interviews and observations to explore the user's perceptions of their own responsiveness and their expectation of responses from other users. They distinguish response expectation from breakdown perception. The former is the implicit time the sender gives to the recipient to respond, which is usually based on the time it took in previous interactions with the recipient. The latter is the initiation of a follow-up action, that occurs when the response expectation time has ended, which is dependent on the recipient, the recipient's location, the topic urgency and whether a voice mail was sent. These findings suggest that the social context of a message might be more important than the contents of the message.

In a survey study with 124 participants, Dabbish et al. (2004) and Dabbish et al. (2005) investigated what characteristics of email messages predict user actions on messages. The authors present a model of reply probability based on the outcomes of this survey. Important factors were the importance of the message, number of recipients, sender characteristics and the nature of the message content. Sender characteristics seemed to have the greatest effect. They did not find an effect of communication frequency on reply probability and suggest that this may be due to the availability of other communication channels that reduce the necessity for email re-

sponse. The perception of message importance was influenced by (1) communication frequency in combination with the status of the sender,(2) whether the message contained a status update, and (3) whether the message was a scheduling event.

There has been several attempts to automatic reply prediction. Dredze, Blitzer, and Pereira (2005) developed a logistic regression predictor that indicates whether email messages necessitate a reply. Their predictor was evaluated on the spam-free inbox and sent-mail folders of two graduate students. Features used were word identity, message length, whether the message contained the mentioning of a date and time, whether the recipient was directly addressed, whether it contained a question and who the recipients or sender was. ROC curves of the trained logistic regression model revealed that to achieve 80 % true positives (message predicted to receive a reply that were actually replied to) there were 50% false positives (message predicted to receive a reply that were not replied to)

In later research Dredze et al. (2008) used a rule based system to predict reply labels (needs reply, does not need reply). In this system they used relational features that rely on a user profile which included the number of sent and received emails from and to each user as well as the user's address book, a supervisor-role indication, email address and domain. Document-specific features were the presence of question marks, request indicators such as question words (weighted using tf-idf scores), presence of attachment, document length, salutations, and the time of day. The system was tested on 2,391 manually labelled emails, coming from 4 students. On average it obtained a precision of 0.73 and recall of 0.64.

In larger scale research using the Enron corpus (Klimt and Yang, 2004; Bekkerman, 2004), Deepak, Garg, and Varshney (2007) and On et al. (2010) investigate the responsiveness and engagingness of users. Their models are based on the number of received replies and the number of sent replies as well as the time it takes to reply. They do not take any content into account.

Ayodele and Zhou (2009) use the Enron corpus to develop and evaluate a manual rule-based reply prediction method. They use largely the same features as Dredze et al. (2008) In a second approach they use only the presence of certain words, salutations, question marks, dates or month names and AM or PM. For both approaches the authors report to have very high accuracies of 100% and 98%. These results are unrealistically high because the e-mails are evaluated manually by human reviewers using the described rules.

In more general research, Aberdeen, Pacovsky, and Slater (2010) try to predict the probability that the user will interact with an email (i.e. open, read, forward or reply) within a certain time span. They use a linear regression model and a form of transfer learning to determine a ranking of the interaction likeliness. A threshold determines which messages are indicated as important.They have used social features (based on interaction with recipient), content features (headers and recent terms that are highly correlated with actions on a message), thread features (interaction of the user with a thread) and label features (labels applied to the message using filters). They obtained an accuracy of 80%. Their work is the basis for the Google Priority Inbox.

7.3. METHOD

The goal of this experiment is to assess whether textual content features have added value when it comes to predicting whether a message will receive a reply or not. For that purpose we select textual features using various feature selection methods (described in Section 7.3.2). We analyse the selected features on their transparency (i.e. how easy are they to interpret?) and evaluate their effectiveness in a classification experiment (Section 7.3.3). We start this section with the description of how we obtained our labelled dataset.

7.3.1. EXTRACTING THREADS FROM THE ENRON CORPUS

To obtain a labelled dataset, we constructed threads from the Enron corpus to determine which message had received a reply and which not. We have used the August 2009 version of the corpus without file attachments to have a fair comparison with the existing literature. We have taken a tree based approach (Venolia and Neustaedter, 2003) for extracting the threads from Enron using the algorithm suggested by Deepak, Garg, and Varshney (2007). From these threads we derived which messages were replies by matching the subject lines and including the prefix "RE:" (case-insensitive). For each reply message we found the corresponding original message (i.e. the message that was replied to) by selecting the message with the same thread id, of which the sender was a receiver of the reply and which was sent before the reply. In the rare case that there were multiple options, we chose the message that was closest in time to the reply. Out of the 252,759 messages in the Enron corpus, we found 3,492 messages that have received a reply and 166,572 message that have not received a reply. We do not take into account messages that are forwards or replies on replies.

7.3.2. FEATURE SELECTION

We have used three different methods for analysing the influence of the textual content of the messages. The first measure is χ^2 (Yang and Pedersen, 1997), which measures the dependence between a term and a class. We are looking for the terms with a high dependency on the replied-class (i.e. a high χ^2 score).

$$\chi^2(t,c) = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{7.1}$$

where A is the number of *replied* messages that contain term t, B is the number of *non-replied* messages that contain t, C is the number of *replied* messages that do not contain term t and finally, D is the number of *non-replied* messages that do not have t. N is the total number of messages in the set.

The second method we used is point-wise Kullback-Leibler divergence (Kullback and Leibler, 1951), as suggested by Carpineto et al. (2001). This measure determines a value for each term which indicates how good that term is for distinguishing the set of replied messages from the set of non-replied messages.

$$KLdiv(t,p) = P(t|p)log\frac{P(t|p)}{P(t|n)} \tag{7.2}$$

where $P(t|p)$ is the same as A from equation (7.1) and $P(t|n)$ is the same as B.

The third and final method is based on linguistic profiling as proposed by Van Halteren (2004). It compares the normalized average term frequency of a term in the positive set (replied messages) to its average in the negative set (non-replied messages). Rather than using the proposed classification method, we use linguistic profiling for term selection.

$$LP(t) = \mu(t,p) - \mu(t,n) \tag{7.3}$$

where $\mu(t,p)$ denotes the normalized average frequency of term t in the set of replied messages and $\mu(t,n)$ denotes the normalized average frequency of term t in the set of non-replied messages.

With all three methods, we extracted the most important terms from the example set. As terms, we considered all word n-grams with $n \in 1,2,3$. For each message, we index the number of occurrences of each term.

7.3.3. CLASSIFICATION

In a classification experiment, we compare the effectiveness of the feature selection methods from the previous section to the effectiveness of the features described in literature. These features (referred to as non-textual features) are: (1) number of receivers in the fields TO, CC and BCC respectively, (2) number of question marks, (3) number of previously received replies from recipient (4) likeliness of interaction with receiver (5) message length (6) occurrence of each of the question words *what, when, where, which, who, whom, whose, why* and *how* weighted with tf-idf . For each of the textual feature selection methods, selections of 10, 50, 100, 500, 1000 and 5000 features were compared.

The original distribution contains 97% negative examples (non-replied e-mails), which is very imbalanced. Therefore, we first rebalance our data by selecting two random negative examples for each positive example in our data. We split our data into 90% train (10476 examples) and 10% test (1167 examples). All examples in the test set have later dates than the examples in the train set to prevent leaking of future information in the training data. We used a Naive Bayes classifier and a J48 decision tree classifier from the WEKA toolkit (Hall et al., 2009), with their default settings. Typically decision tree works well for non-text features and Naive Bayes is well-suited for textual features. The WEKA re-sample filter is used to balance the data uniformly by oversampling the positive examples. The reason for first under-balancing the negative examples is to prevent a too extreme oversampling of the positive examples. The results were evaluated on the fixed unbalanced test set.

7.4. RESULTS

7.4.1. FEATURE ANALYSIS

The top 50 n-grams, of which 10 are presented in Table 7.1, of each of the three feature selection methods were manually analysed. Both the point-wise Kullback Leibler and the Linguistic Profiling method indicate the importance of the personal pronouns *I, we* and *you*. These pronouns may indicate that the receiver is addressed personally. All methods also seem to indicate the occurrence of the phrase “please let

Table 7.1: Top 10 n-grams that indicate that a message will receive a reply

$\chi^2(t,c)$	$KLdiv(t,p)$	$LP(t)$
me	i	fyi
keep	we	i
i	me	me
2001	you	we
information	have	you
decisions	know	know
one of	let	have
let	please	http
news	let me	let
receive a	me know	2001

me know" which suggests that the sender expects an action from the receiver. Worth noting is that Linguistic Profiling indicates the importance of the term "fyi". Even though this does not seem intuitive, inspection of messages reveals that "fyi" messages often receive a "thank you" reply. The terms selected by the χ^2 measure seem to be less easy to interpret. They may refer to more specific situations. Overall, first analysis suggests that of the top 50 terms point-wise Kullback Leibler term selection is the easiest to interpret and the least sensitive to noise.

7.4.2. CLASSIFICATION

Table 7.2: Classification results for the optimal number of features. Reported precision and recall are for the "will reply" class only. Best results are indicated in bold face. BOW refers to a full bag of words frequency model (no selection)

		Naive Bayes		
Feature Type	# Features	Accuracy	Precision	Recall
non-Text		42.6%	0.358	**0.912**
$\chi^2(t,c)$	1000	59.0%	0.43	0.709
$KLdiv(t,p)$	10	70.8%	**0.635**	0.291
$LP(t)$	50	**72.0**%	0.586	0.544
BOW	117400	58.7%	0.417	0.611
		Decision Tree		
Feature Type	# Features	Accuracy	Precision	Recall
non-Text		69.9%	0.581	0.351
$\chi^2(t,c)$	10	66.9%	0.502	0.557
$KLdiv(t,p)$	500	65.7%	0.475	0.291
$LP(t)$	50	64.1%	0.441	0.296
BOW	117400	62.2%	0.432	0.436

Table 7.2 shows the classification results for the optimal number of features with the various feature selection methods and the two classification approaches. The reported precision and recall are for the "will reply" class only.

When we look at the Naive Bayes results in Table 7.2 we see that if we select as little as 50 features from the LP measure we have a reasonable accuracy (72.04%). The

classifier with only non-textual features, performs much worse and shows an accuracy of 42.62%. Interestingly its recall for the positive class is very high: it recognizes more than 90% of the emails that received a reply.

When we look at the results for the decision tree, we see that the classifier with non-textual features performs better than with Naive Bayes (69.98%), while the runs on only textual features selected by $\chi^2(t,c)$, $KLdiv(t,p)$ and $LP(t)$ all give an accuracy around 65%. Interestingly, $\chi^2(t,c)$ performs a lot better than in the Naive Bayes classifier, while $KLdiv(t,p)$ and $LP(t)$ perform worse. We only found very small differences in classification performance when we vary the number of selected features.

Combined classifiers that were trained on the combinations of textual and non-textual features performed approximately as good as the best classifier of the two.

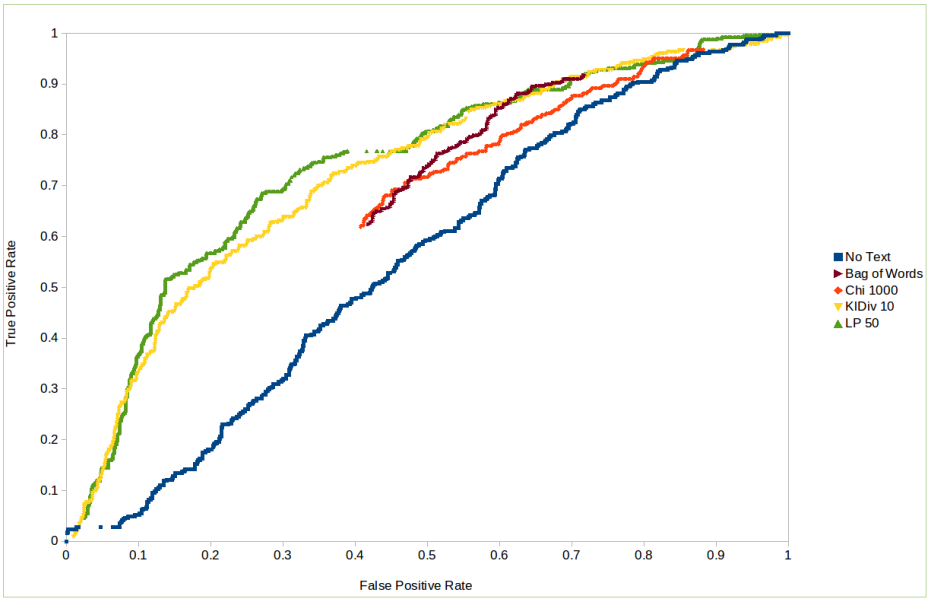


Figure 7.1: ROC curves of term selection methods with a Naive Bayes classifier

It is interesting to notice that the performance of the feature selection method $\chi^2(t,c)$ is so different with the two classifiers. $\chi^2(t,c)$ is often used as a feature selection method for text classification, especially in Naive Bayes, while this experiment suggests that point-wise Kullback-Leiber divergence and Linguistic Profiling might be better feature selectors.

This is confirmed when we look at the ROC curves of the Naive Bayes classifier (Figure 7.1). We see that the curves for Linguistic Profiling and Kullback-Leibler divergence are very similar and that their AUC-values are higher than for $\chi^2(t,c)$ and bag-of-words (BOW). It is also clear from the curves that the text conditions have higher AUC-values than the non-text condition in a Naive Bayes classifier.

7.5. CONCLUSION

In the current work we found that after first analysis of three feature selection methods for reply prediction, point-wise Kullback Leibler divergence seems a useful measure to select interpretable terms that are representative of a class. Linguistic Profiling seems suitable as well but seems to contain a little more noise.

Using a Naive Bayes classifier we only need as little as 50 terms selected by linguistic profiling to achieve a reasonable accuracy (72.04%). This is even better than our baseline results with non-text features with a decision tree (69.98%), but only slightly. On the other hand, we obtained the highest recall with non-text features.

Concluding, we can predict with reasonable accuracy which e-mails will be replied to. Although 72% success might not be accurate enough to be used as a stand-alone application, we can use it as an indication of how important that message is. However, whether a message will be replied to is likely not the only determinant of message importance, so future work may include other methods for estimating message importance.

Additionally, transparency is an important concept in SWELL, and we think that it is important to find a good balance between precision and recall, so that the user has trust in the system (i.e. does not feel like important messages are missed), but also understands why some indications are given, and does not require too much additional feedback. Given the results of our experiment it seems important to find a method that combines a classifier with high recall such as Naive Bayes with non-text features, and a classifier with high precision such as Naive Bayes with features selected by Kullback-Leibler divergence.

8

E-MAIL CATEGORIZATION USING PARTIALLY RELATED TRAINING EXAMPLES

Edited from: **Maya Sappelli, Suzan Verberne, Wessel Kraaij** (2014) *E-mail categorization using partially related training examples*, Proceedings of the 5th Information Interaction in Context Symposium (IIiX 2014).

Automatic e-mail categorization with traditional classification methods requires labelling of training data. In a real-life setting, this labelling disturbs the working flow of the user. We argue that it might be helpful to use documents, which are generally well-structured in directories on the file system, as training data for supervised e-mail categorization and thereby reducing the labelling effort required from users.

Previous work demonstrated that the characteristics of documents and e-mail messages are too different to use organized documents as training examples for e-mail categorization using traditional supervised classification methods.

In this chapter we present a novel network-based algorithm that is capable of taking into account these differences between documents and e-mails. With the network algorithm, it is possible to use documents as training material for e-mail categorization without user intervention. This way, the effort for the users for labelling training examples is reduced, while the organization of their information flow is still improved.

The accuracy of the algorithm on categorizing e-mail messages was evaluated using a set of e-mail correspondence related to the documents. The proposed network method was significantly better than traditional text classification algorithm in this setting.

8.1. INTRODUCTION

The life of knowledge workers is changing rapidly. With the arrival of mobile internet, smart phones and the corresponding "any place any time information" it be-

comes increasingly hard to balance work life and personal life. Additionally, knowledge workers need to be able to handle large amounts of data. Receiving more than 70 new corporate e-mail messages a day is not uncommon (Radicati, 2010) so an effective personal information management system is required to be able to organize and re-find these messages. For this purpose 'working in context' is deemed beneficial (Gomez-Perez et al., 2009; Warren, 2013). Assistance of knowledge workers with 'working in context' is one of the goals of the SWELL project[1] for which this research is executed.

One application area of interest is the e-mail domain. Associating e-mail messages with their contexts has two benefits: 1) it can help knowledge workers find back their messages more easily and 2) reading messages context-wise, for example by project, is more efficient since the number of context switches is minimized. This latter aspect is a suggestion from the 'getting things done' management method (Allen, 2003).

Many e-mail programs have an option to categorize or file messages, which allows for the possibility to associate messages with for example a 'work-project' context. This categorization option however, is often not used optimally, as messages are left to linger in the inbox (Whittaker and Sidner, 1996) and many users do not even use category folders at all (Koren et al., 2011). Manually categorizing the messages is too big an effort for busy knowledge workers, diminishing the actual benefits of the categorization.

Automated approaches for e-mail message classification are plentiful. The early work in e-mail classification was mostly directed towards detecting spam (Sahami et al., 1998). This was followed by work towards categorizing e-mails in order to support personal information management (Segal and Kephart, 1999; Bekkerman, 2004). Nowadays, work on classifying e-mails is often directed towards predicting the action required for the message (Dredze et al., 2008; Aberdeen, Pacovsky, and Slater, 2010; Sappelli, Verberne, and Kraaij, 2013a). Only automatic spam classification has become a commodity in email handling. Categorization functionality within e-mail clients often relies on hand-crafted rules.

The downside of the methods based on machine learning is that each of them still requires labelled training data. Although this training dataset only needs to be a limited but representative part of all messages, it still requires effort from the knowledge worker as they would need to label these examples. Especially the persons that receive the most messages, and will most likely benefit the most from a good categorization, will probably not have the time to provide a sufficient amount of labelled examples. Furthermore, knowledge workers are often not consistent in their categorizations (Whittaker and Sidner, 1996).

Sappelli, Verberne, and Kraaij (2012) tried to reduce the effort required from users by using existing folder structures and the documents in them as training material for supervised algorithms to classify unstructured e-mail data. This is motivated by the idea that especially in a work setting, projects are often organized in project folders. These projects would function as an intuitive context of e-mail messages and therefore would be good categorizations. However, the study demonstrated that the

[1] www.swell-project.net

characteristics of documents and e-mail messages are too different to use organized documents as training examples for e-mail categorization using traditional classification methods.

The goal of the research presented in this chapter is twofold: (1) we aim to improve upon the work by Sappelli, Verberne, and Kraaij (2012) and (2) we aim to evaluate our new supervised network-based classification method. Since traditional methods such as K-Nearest Neighbours, Naive Bayes and SVM proved unsuccessful when presented with training materials of a different type than the test data, we have developed a method that combines the specific characteristics of documents and e-mail messages and exploits these characteristics to make a more robust classifier.

In Section 8.2, we describe some supervised, unsupervised and semi-supervised approaches to e-mail categorization. The new network-based classification method that is proposed is described in Section 8.3. In Section 8.4 the evaluation of this algorithm is described, followed by a discussion and our conclusions in sections 8.5 and 8.6.

8.2. RELATED WORK

Methods for classification can be divided into supervised approaches, where training examples are provided, and unsupervised approaches, where there is typically no training involved. There are also semi-supervised approaches, where a combination of labelled and unlabelled data is used to reduce the training effort compared to supervised methods. A specific form of semi-supervised learning is transductive transfer learning (Bahadori, Liu, and Zhang, 2011), where knowledge from one domain is transferred to another domain, and where the source domain has an abundance of labelled examples while the target domain has none. This is the approach we take in the presented algorithm in this chapter. In this section we describe a few typical e-mail categorization methods in each of the categories.

8.2.1. SUPERVISED CATEGORIZATION

Although supervised machine learning methods require labelled data, which implicates that they need input from the user, this is the main approach for e-mail classification. Various machine learning algorithms have been proposed. For example, Segal and Kephart (1999) use a classifier in their e-mail organization program MailCat that uses the similarity between a word-frequency vector of a message and TFIDF weighted vectors of categories to determine the correct category. Their algorithm achieves 60-80% accuracy.

Bekkerman (2004) evaluate Maximum Entropy (ME), Naive Bayes (NB), Support Vector Machines (SVM) and Winnow on the classification of two e-mail datasets, among which the Enron dataset. Overall, SVM has the highest performance (55-95% dependent on the persons whose messages are categorized).

On the other hand, Chakravarthy, Venkatachalam, and Telang (2010) provide a graph-based approach to email classification which they also evaluate on the Enron dataset. Their performance varies with the number of classes that need to be recognized (60-90% accuracy)

Krzywicki and Wobcke (2010) present a method for incremental e-mail categorization. This is based on the idea that the categories in a changing dataset like e-mail change over time. New topics are introduced and older topics can become irrelevant. Their 'clumping' method looks at local coherence in the data. They evaluate their results on the Enron dataset and obtain comparable results as SVM on that dataset (58-95%). Their method however is less complex and therefore has a lower execution time.

Interestingly the variation in classification accuracies presented in existing literature is large. Furthermore, each of these methods requires a large dataset. Usually 70-80% of the data is used as training material. For the 7 largest users in the Enron dataset this corresponds to more than 2000 messages on average that need to be labelled.

8.2.2. UNSUPERVISED CATEGORIZATION

In unsupervised machine learning methods, usually clustering techniques are used. Xiang (2009) presents a non-parametric clustering algorithm using Hubert's γ. They report an accuracy of 70% on average, measured on two personal datasets whereas K-means achieves 47% and Hierarchical Agglomerative Clustering obtains 60% accuracy.

Furthermore, Kulkarni and Pedersen (2005) present their system Sense-Clusters. The authors see an e-mail message itself as a context and seek to group these contexts. Grouping is based on the similarities in content using occurrence and co-occurrence matrices. Labels are given using descriptive and discriminating terms in the clusters. They test their algorithm on the 20-NewsGroups Corpus and report a F-score of 61–83%. The quality of the labels is not evaluated.

These performances are comparable to the supervised setting, and these approaches require no labelling effort of the user. However, there is still an open issue as sometimes the clusters are not labelled or the labels might not be meaningful enough to the user. Additionally, the clustering is based on similarities between messages, and it is by no means certain that these clusters are the clusters the user is looking for.

8.2.3. SEMI-SUPERVISED CATEGORIZATION

There are some approaches that try a combination of supervised and unsupervised learning. Kiritchenko and Matwin (2001) try to reduce the number of required training examples for SVM and Naive Bayes by using co-training. In this technique they separate the features in 2 sets, and train one weak classifier on one set and one on the other. For new examples, each classifier labels the example, and the most confidently predicted positive and negative examples are added to the set of labelled examples. In essence the two classifiers train each other, since when one classifier is confident about a new example, this information can be taught to the other classifier. The results show that this technique can improve SVM classifiers, but also that it has a negative impact when using Naive Bayes classifiers.

Huang and Mitchell (2008) propose a mixed-initiative clustering approach for e-mail. The algorithm provides an initial clustering of the messages and the users can

iteratively review and edit the clustering in order to constrain a new iteration of automatic clustering. The required effort from the user is halved using this approach, but no interpretative labels are provided for the clusters.

In the e-mail classification method by Park and An (2010) the categories result from clustering, but category labels are obtained from a set of incoming e-mails using either latent semantic analysis or non negative matrix factorization. When users are unsatisfied with the category hierarchy derived from the semantic features, they can opt for a dynamic category hierarchy reconstruction which is based on fuzzy relational products. Park et al. did not test their algorithm on an e-mail dataset, but rather on the Reuters document collection. Also, they did not evaluate the quality of the labels that their algorithm provides. It is possible that although their approach is interesting, it might not work as well on e-mail compared to the documents of the Reuters corpus, considering the differences that Sappelli, Verberne, and Kraaij (2012) have found. Moreover, it is not certain that the category labels proposed by the algorithm are meaningful to the user.

Two transductive-transfer learning examples come from Koren et al. (2011) and Sappelli, Verberne, and Kraaij (2012). Koren et al. (2011) propose to use other user's folders to suggest categories, such that users that do not have time to categorize messages themselves can benefit from the categories that others make. Although this would be a solution to reduce the effort for some people, it would require the access to data of other users, which poses serious privacy issues. Also, it would be much harder to use social features such as sender and receiver, since it is unlikely that multiple users have the same social dynamics.

Sappelli, Verberne, and Kraaij (2012) compared traditional supervised algorithms such as K-nearest neighbours, SVM and Naive Bayes on the task of e-mail categorization, but provided categorized documents as training data instead of e-mail messages. They tested the algorithms on a personal set of e-mail messages. The authors found that the algorithms were not successful in categorizing e-mail messages when they were trained on related documents. An analysis of models trained on e-mail messages showed that the features required for successfully categorizing messages (such as names and addresses) are too different from the features that are extracted from the categorized documents (content words in general). In fact, the documents do contain the features that are needed for the categorization of e-mail messages (e.g. in the form of author names), but the traditional classification methods are not successful in extracting these features.

8.3. OUR MODEL FOR E-MAIL CATEGORIZATION

In Sappelli, Verberne, and Kraaij (2012), the authors found that common machine learning algorithms such as Naive Bayes, K-Nearest Neighbours (k-NN) and SVM are not successful in using documents as training examples for classifying emails. The main reason is that for email categorization, contact details, such as the sender or recipient of a message, are the most distinguishing features, while for documents the topic is much more important. In Naive Bayes, k-NN and SVM there was no distinction between the type of features as they were all uni-gram (bag-of-words) based.

We propose to bridge the domains of emails and documents by introducing con-

tact names as an additional category of features for the joint space of documents and emails. Our model can be viewed as a transfer learning approach where labelled data in the domain of documents is used to learn a classifier for emails, for which no labelled training items are available. In the approach proposed in this chapter, the contact-type features in the data play an important role. In e-mail messages, these contact type features are often e-mail addresses, while in documents these features are usually (author) names. To connect the names and contacts, together with other information, we use a network based approach which is based on the interactive activation model by McClelland and Rumelhart (1981). The original interactive activation model is a cognitive model based on theories on neural activity in the brain. The idea is that the model consists of nodes and connections and that activation spreads through the network to activate other nodes. These nodes are comparable to neurons in the brain and the notion of spreading activation is similar to neurons transmitting electrochemical signals to each other. As in the brain, nodes in the model can send inhibitory or excitatory signals. Where this model was originally used to assess validity of cognitive theories, it can also be used as a method for context recognition (Verberne and Sappelli, 2013). In contrast to typical neural networks, there are no hidden layers in the model for context recognition and the connection weights are not learned.

There are two phases in the interactive activation approach. First the network with nodes and connections need to be constructed (Section 8.3.1). Secondly, to obtain a classification for an input, the activation needs to be spread through the network (Section 8.3.2).

8.3.1. CONSTRUCTING THE NETWORK

The network consists of 3 layers as depicted in Figure 8.1:

- the input layer; the e-mails that need to be categorized
- the context information layer; the various elements that can be extracted from e-mails and documents. These tell us something about the context of the message or document. For the current problem we focus on social context (person names), topics or terms and location information. These information types were chosen because they have a relation to both e-mail messages and documents.
- the output layer; the categories that the user is interested in

Each of these layers consists of nodes. Each node can have one or more weighted connections to other nodes.

First, the contact nodes in the context-information layer are created. We do this by using the knowledge worker's address book on the computer. Names are divided into first names and last names. Only the actual names are kept, words like 'van de' are removed. Each first and last name and each e-mail address receives a node in the context information level. Names and addresses are connected using the address book. Names can be associated with multiple addresses. For example 'John Doe' is divided into 'John' and 'Doe'. The name is associated with

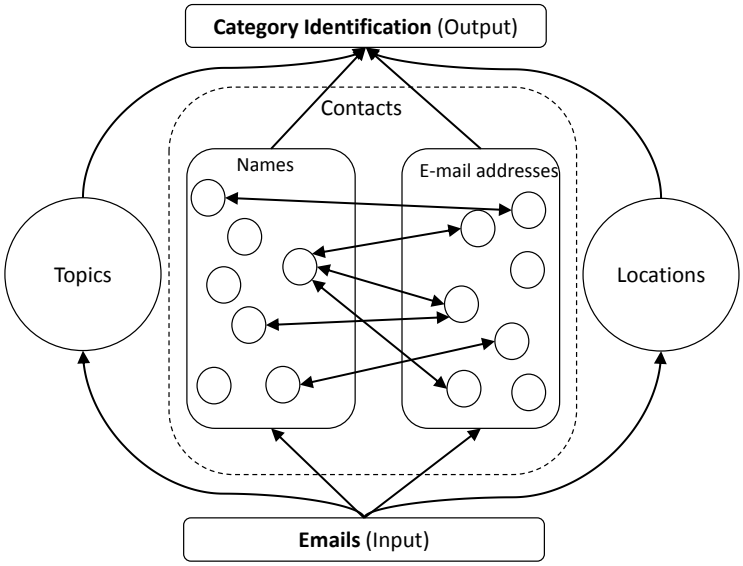


Figure 8.1: Representation of the network for category identification

'john.doe@email.com'. Both the node 'John' as well as the node 'Doe' are connected to 'john.doe@email.com'. If there is also a person named 'John Peters' with the address 'myaddress@email.com' in the addressbook, the node 'john' is connected to the node 'myaddress@email.com' as well. Furthermore, e-mail addresses can be connected to multiple nodes as well, for example when an e-mail address is shared between 2 persons. This idea of connecting nodes is cognitively plausible: When you just read the name 'John' you do not know yet whether this refers to 'John Doe' or 'John Peters' and thus you can think of both e-mail addresses as possible senders.

Additional context information can be added during this phase as well. For constructing the topic nodes, terms or phrases in the documents that are descriptive of the categories are extracted using a weighted combination of three term scoring methods as described in Verberne, Sappelli, and Kraaij (2013). The 100 most important terms for a category are extracted, resulting in a set of 19658 important terms and phrases for all categories together when duplicates are removed. These terms and phrases are interesting as they could contain project names, but in any case they represent the content of the documents.

Another type of context information are the location nodes. For each document the filename without the path is extracted and added as a location node. The motivation for these nodes is based on the possibility of attaching files to messages. Since these attachments do not have a path, only the filename itself is informative. If one of these file attachments matches a filename in the location nodes, this is strong evidence for the category of the document from which the location node is created.

The connections from the nodes in the context information layer to the category nodes in the output layer are made based on the analysis of documents on the user's computer. The connections are made by creating a category node in the output layer for each of the project-folders that the user selects. This selection of relevant folders is the only 'supervision' that is required from the users. After the creation of these category nodes, they are connected to the context information layer by analysing the documents in the corresponding project folder. From each document, names and e-mail addresses that occur in the context information layer are extracted. Each name or e-mail address that occurs is connected to the category name that the document belongs to. The same process is repeated for the terms. If one of the important terms is found in a document, then a connection is made from the term to the category of the document. For the locations, the documents do not need to be analysed, simply a connection can be made from the location node to the category of the document from which the location node was originally created.

The connections from input layer to context information layer are created during run-time.

Each connection from node n1 to node n2 has an inverse document frequency style connection weight:

$$\frac{1}{\#outputconnections_{n1}} \tag{8.1}$$

This means that if a node has only a few connections and it is activated it has a high impact, but if the node is connected to many other categories it becomes less important.

For the purpose of the experiments in section 8.4 the identification layer and the context information layer are fixed at this point; only connections between input and context information layer are added during run-time. However, in a learning setting as presented in section 8.4.3.2, the network can be adapted further as new connections between context information layer and identification layer can be made given the input message and the corrected or confirmed category. Additionally, it would be expected in a real application that the network is updated regularly, to allow new categories or remove obsolete categories.

8.3.2. RUNNING THE NETWORK

To obtain a category for an input the network needs to be activated. First the e-mail message is added to the input layer as a node. Next, the names, e-mail addresses, topics, file attachment names and other potential sources of information are extracted from the message and connections to the corresponding nodes are created. Then the activation of the input node corresponding to the message is set to 1.0.

First, the weighted excitatory input ex and weighted inhibitory input in are calculated. These are weighted sums of each of the excitatory or inhibitory input connections to a node:

$$ex_j = \sum_{c_{i,j} \in C_{excitatory}} \alpha \cdot a_i \cdot w_{c_{i,j}} \tag{8.2}$$

where $c_{i,j} \in C_{excitatory}$ is a connection in the set of input connections to node j where $a_i > 0$: the activation of the from-node i is greater than 0. α is the parameter

for the strength of excitation and $w_{c_{i,j}}$ is the connection strength between node i and j.

$$in_j = \sum_{c_{i,j} \in C_{inhibitory}} \gamma \cdot a_i \cdot w_{c_{i,j}} \tag{8.3}$$

where $c_{i,j} \in C_{inhibtory}$ is a connection in the set of input connections to node j where $a_i <= 0$. γ is the parameter for the strength of inhibitions.

Activation of each of the nodes in the network is updated using Grossberg's activation function (Grossberg, 1976):

$$\delta a_j = (max - a_j)ex_j - (a_j - min)in_j - decay(a_j - rest) \tag{8.4}$$

where a is the current activation of a node, ex is the weighted excitatory input of the node (8.2), in is the weighted inhibitory input (8.3) and min, max, $rest$ and $decay$ are general parameters in the model (see also Table 8.1). This function ensures that the activation of a node will go back to the resting level when there is no evidence for that element, and towards 1.0 when there is a lot of evidence for that element.

Normally the network would be run for the number of iterations required to stabilize the activation in the network. However, for pragmatic reasons the network is run for 10 iterations for each input message. This is enough to activate the network properly (i.e. activate all levels) and keeps the running time low. This would be a realistic requirement when the algorithm would be put to use in an actual application. Moreover, more than 10 iterations did not improve accuracy in the experiments described in section 8.4. This suggests that the network stabilizes quickly.

To obtain the label for the input message, the activation of the category nodes in the output layer can be read. Each node starts with the same resting level, but the variation in number of input connections to a node together with the excitation and inhibition parameters can alter this resting level slightly. Therefore, the increase in activity of a node is compared to its individual start level and the node with the highest increase in activation will be selected as label for the input message.

8.4. EXPERIMENTS

In a series of experiments we compare our method for e-mail categorization using documents as training data to the previous approach by Sappelli, Verberne, and Kraaij (2012). In the first experiment we look at a network with only contact nodes. In a second and third experiment we add topic nodes and location nodes respectively to see whether this enhances the network.

8.4.1. DATA

We obtained the personal email and document dataset from Sappelli, Verberne, and Kraaij (2012). This dataset consists of 354 documents and 874 e-mails.The documents as well as the emails were provided in raw text form. This data had been manually categorized into 43 categories corresponding to 43 different courses followed by the single student who provided the dataset. These courses were followed in 4 years time and are part of 2 different curricula; Linguistics and Artificial Intelligence (AI). A third curriculum-type category was the Thesis category, as this was a combination

of both the Linguistics and AI curricula. The data is hierarchically ordered based on curriculum and year (See Figure 8.2).

Our aim is to support knowledge workers in their working life by categorizing messages to projects. Although at first sight a dataset of a student's course related documents and e-mail messages might not seem relevant for the knowledge worker's life, there is a clear link. Both courses and projects have contextual elements. They both have topics, they both have documents related to the topics, and in both projects part of the work or all the work can be executed in collaboration. Thus, both courses and knowledge worker projects have a social and topical context. In fact, a course can be seen as a project that the student is working on.

There are three relations between the documents and e-mails in the dataset. First, the documents and the messages are about the same courses, so there is a topical relation between them. Furthermore, the documents and messages share a time relation as the messages are sent and received during the course period. Some of the documents have been created by the student in that time period as well, while some documents, such as course materials written by the teacher, have already been obtained at the start of the course. This means that the training documents do not necessarily precede all e-mail messages in their creation dates. Finally, documents can be send via e-mail messages, creating an attachment relation between a message and a document. However this last relation was not very common as only 5.8% of the e-mail messages contained a training document as file attachment.

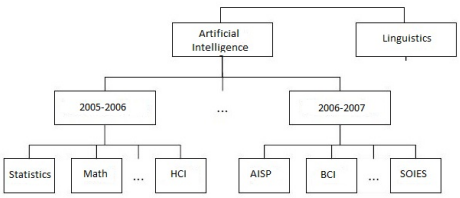


Figure 8.2: Example of document category structure (Adopted from Sappelli, Verberne, and Kraaij (2012))

8.4.2. PARAMETER OPTIMIZATION

A small subset of the data (7 categories, 122 documents, 53 e-mail messages) was used to optimize the parameters in our network. A grid search optimization was executed with the minimum, maximum and step-sizes as mentioned in Table 8.1.

Table 8.1 also shows the default parameters of the original IA model, and the parameters that we used in the experiment. There were multiple sets of parameters that proved to be optimal. We have chosen to use a set that seemed logical. The strength of excitation in the network was increased while the strength of inhibition was decreased compared to the default. This boosted the impact of observed nodes, while reducing the effect of unobserved nodes. The decay parameter was also increased. The decay parameter pushes the activation back to the resting value, which happens faster with higher decay values. This was as expected for categorizing e-mail mes-

Table 8.1: Parameters of the network

Parameter	Definition	Min	Max	Stepsize	Default	Optimal
α	Strength of excitation	0.0	1.0	0.1	0.1	0.2
γ	Strength of inhibition	0.0	1.0	0.1	0.1	0.0
Min	Minimal value of activation	−1.0	0.5	0.1	−0.2	−0.2
Max	Maximal value of activation	0.0	1.0	0.1	1.0	1.0
$Rest$	Resting-level of activation	−1.0	0.5	0.1	−0.1	−0.1
$Decay$	Strength of decay	0.0	1.0	0.1	0.1	0.3

sages since there does not need to be a relation between one message and the next.

8.4.3. RESULTS

In Table 8.2 we present the accuracy of the presented network method in various forms. We compare the accuracy of the algorithm presented in this chapter to traditional algorithms such as Naive Bayes, K-nearest neighbours (k-NN) and Linear SVM. These are the baseline runs and are in the top part of table 8.2. These traditional algorithms are trained on a bag of word uni-gram model with TF-IDF weighting, where k-NN and SVM are pruned: words that occurred in less than 3% or more than 30% of the documents were excluded from the feature vectors. The accuracy that can be obtained when the most frequent class is always selected is also presented (ZeroR). The reported accuracies are adopted from Sappelli, Verberne, and Kraaij (2012).

In addition we improved the term selection for Naive Bayes, k-NN and linear SVM by using the same term extraction method as in the network approach, which is described in Section 8.3.1.

The significance of the difference between the algorithms and the best baseline run (k-NN) is measured using McNemar's test (Dietterich, 1998). This statistical test measures whether the marginal frequencies are equal and can be used on paired nominal data. The null hypothesis is that two classifiers (C1 and C2) are equally accurate. The McNemar test statistic is:

$$\chi^2 = \frac{(b-c)^2}{b+c} \tag{8.5}$$

where b is the number of times C1 is correct and C2 is wrong, and c is the number of times C2 is correct and C1 is wrong. The χ^2-value is compared to the chi-squared distribution to determine the p-value.

When we look at the result of a network with only topic nodes, it already has a significantly higher performance than Naive Bayes ($p < 0.001$), while the result is comparable to SVM ($p = 0.39$). The network approach uses less features than Naive Bayes.

When looking at location nodes (i.e. filenames matching file attachments), the accuracy of the network goes towards the majority baseline and is a significant improvement over Naive Bayes ($p < 0.001$) as well as SVMs ($p < 0.001$).

Using only contact nodes in the network gives a significantly better performance ($p < 0.001$) than the traditional classification methods that were explored so far, even

Table 8.2: Classification Accuracy. The top block shows the baseline accuracies adopted from Sappelli, Verberne, and Kraaij (2012). The middle block shows the accuracies obtained with the network method. The bottom block shows improved accuracies for the supervised methods using either the same term selection as in the network (descriptive term extraction) or contact names

Classifier	Accuracy
ZeroR (majority baseline)	15.3%
Naive Bayes (no pruning)	1.7%
Linear SVM	7.8%
k-NN, K=5	20.9%
Network – Contacts only	53.7%
Network – Topics only	7.1%
Network – Location only	14.0%
Network – Contacts & Location	56.7%
Network – Contacts & Topics	57.7%
Network – Contacts, Location & Topics	**58.3%**
Naive Bayes – Descriptive term extraction	3.2%
Linear SVM – Descriptive term extraction	15.9%
k-NN, K=5 – Descriptive term extraction	32.8%
k-NN, K=5 – Contacts only	54.6%

the ones that were improved by using the descriptive term extraction method as described in Section 8.3.1. However, if we train a k-NN algorithm with $k = 5$ on the documents where we first extract all contact features, we can obtain a similar accuracy of 54.6%. There is no significant difference between k-NN with only contacts and the network with only contacts ($p = 0.54$), but the network has the advantage that additional information types can be added. Adding location nodes to the network with contact nodes boosts the performance a little bit further to 56.7%. A network with contact nodes, location nodes as well as topic nodes gives the highest performance of 58.3%. Although this is a nice improvement, the difference between the complete network and k-NN with only contact nodes is not significant ($p > 0.06$).

INFLUENCE OF NUMBER OF CLASSES

As discussed before, we aim to support knowledge workers by categorizing messages to the projects that the knowledge worker is working on. It is not realistic that a knowledge worker would work on 43 projects at the same time. Therefore we have looked at the influence of the number of categories.

We first try to make the task easier by categorizing the messages to curriculum rather than to course level. This is essentially a categorization higher up in the hierarchy. The curriculum level has 3 categories; *Artificial Intelligence (AI)*, *Linguistics* and *Thesis*. The *Thesis* category is not actually a curriculum, but is rather a combination of both *AI* and *Linguistics* and is therefore placed on the curriculum level. The full network (contacts, topics & locations) correctly classifies 73.5% of the messages. When we look at the confusion matrix in table 8.3 it becomes apparent that most mistakes are made between the *AI* messages and *Thesis*-messages (11% error). This is not strange, as the thesis was a continuation of a course in the *AI* curriculum, so there is a large overlap between contacts.

More interesting is the number of *AI* messages that are mistakenly classified as *Linguistics*, while there are no *Linguistics* messages mistakenly classified as *AI*. It seems that these errors are related to ambiguity in the names. Both categories have connections to a couple of first-names that occur often even though the actual persons to which they refer may be different. The messages that are wrongly classified, typically have these common names associated with them. Apparently the *Linguistics* category is favoured in the case of these ambiguous situations.

Table 8.3: Confusion Matrix Network - Curriculum

	AI predicted	Thesis predicted	Linguistics predicted
AI	421	90	82
Thesis	5	104	26
Linguistics	0	24	104

A k-NN algorithm filtered on contacts actually achieves a significantly higher accuracy of 79.3% ($p < 0.001$) compared to the network algorithm when tested on curriculum categories. The confusion matrix in Table 8.4 shows that for k-NN there is much less confusion between *AI* and *Linguistics*. There are two possible explanations: 1) the impact of ambiguity in names is smaller because the weighting in k-NN is different or 2) the network algorithm is harmed by the influence of previous messages.

Table 8.4: Confusion Matrix k-NN - Curriculum

	AI predicted	Thesis predicted	Linguistics predicted
AI	485	61	47
Thesis	27	100	8
Linguistics	24	10	94

In a second attempt to make the task more realistic, we categorize messages to courses again, but the network is built year-wise, such that there are only 5-14 categories at a time. The training data then consists of only the documents corresponding to the courses that were taken in a specific year.

Table 8.5: Classification Accuracy: year-wise training

Year	#categories	accuracy Network	accuracy k-NN
2005-2006	5	20.5%	73.1%
2006-2007	12	20.0%	22.5%
2007-2008	14	40.8%	31.6%
2008-2009	11	73.6%	62.9%

For the k-NN algorithm trained on contacts only there is a clear advantage of a reduced number of categories (See Table 8.5). Interestingly this does not seem to be

the case for the network algorithm. The total accuracy that could be obtained is for both algorithms lower than when all categories are classified at the same time.

When the accuracies of the network algorithm are analysed, the accuracy for the courses in the year 2008-2009 is the highest. Most likely this is because, 2008-2009 is a particularly easy year as 84.9% of the messages can be classified in 3 out of 11 categories.

Similarly, 2005-2006 seems to be a very difficult classification. In 2005-2006 there are only 5 courses, and in 3 of them there was a collaboration with the same persons. Moreover all 3 courses were quite similar in topic. The most confusion in this year existed between the courses *Datastructures* and *HCI*: 57.1% of *Datastructures* messages were misclassified as *HCI*. There were actually 43 documents for *Datastructures* while there were only 4 for *HCI*. After inspection of these training documents, it is clear that the *HCI* documents contain more social references related to the course, making them more suited as training material for e-mail categorization.

However, this problem would be the same for the k-NN algorithm, but k-NN seems less influenced by the overlap in contacts and the lack of references in some of the training documents. Moreover, k-NN seems better at selecting the larger categories when the input is ambiguous, which improves the accuracy greatly. Nevertheless it has a class recall of 0% for 3 out of 5, which is not satisfactory. We expect that k-NN has a higher accuracy for the year 2005-2006 because the larger categories also had many more documents. This means that there are more documents that can be close to a message and a higher chance that k-NN will select that category as the class. The network is not influenced by the number of documents as there only needs to be one example to make a connection. In this particular year, this has a large influence since the network prefers a smaller class *HCI* over the larger class *Datastructures*, whereas k-NN always prefers *Datastructures*.

LEARNING CURVE

The model can be improved when the content of the e-mail messages is used to make additional connections in the network. Initially there will be no labelled messages available, but as the user uses the system it can correct or confirm categorizations. From these corrections and confirmations the network can learn, because for these messages it is absolutely certain what the label should be. New connections between context information nodes and category labels can be made. In particular, direct connections between e-mail addresses and categories can be created. In the learning curve experiment we look at a model with only contact nodes to see what we can achieve with the least complex network possible.

Figure 8.3 shows the learning curve of the network. Increasingly more labelled e-mail examples, are presented to the network, improving the classification accuracy. For this experiment we chose to randomly select the e-mail examples, as this would be a realistic setting when users confirm or correct labels. The figure shows the learning curve for the situation where an initial model based on documents is improved, as well as a traditional supervised situation where there is no initial model.

From the figure it is clear that the learning curves are steep. The network learns quickly when it is presented with e-mail messages, and stable 80% accuracy is obtained by the network with the initial model at around 20% training examples (about

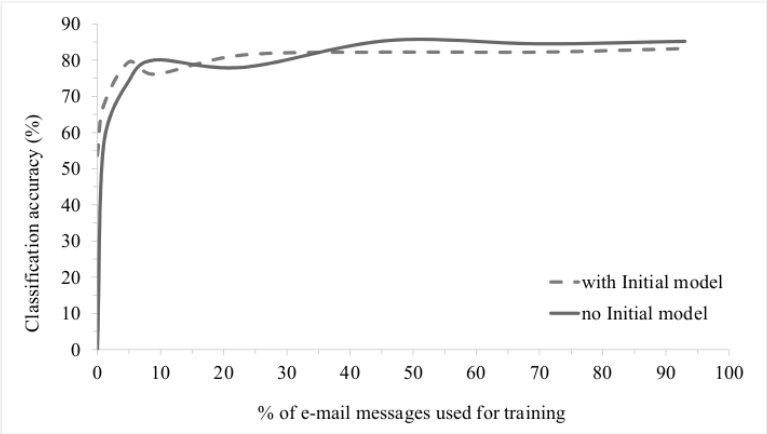


Figure 8.3: Learning curves of the categorization algorithm with initial training on documents vs. without initial training.

170 messages). For the model without the initial training on documents this is obtained at approximately 35% training examples (about 300 messages). Overall, the model without the initial training obtains an accuracy of 85%, which is not significant compared to the 83% accuracy for the model with initial training on documents. The actual selection of training examples has a large impact on the performance. If all training examples that are presented come from the same category, the impact is much smaller, since less new information is introduced. Optimally, at least one example per category should be selected.

The maximum accuracy of the network model is not as high as a supervised Naive Bayes model trained only on emails (89.9%)(Sappelli, Verberne, and Kraaij, 2012). With a Naive Bayes supervised algorithm 20% of training examples are required to achieve 80% accuracy, just like the situation with the initial network model. However, with Naive Bayes, there is actually a decreased performance when a model initially trained on documents is used (30% training examples required and a maximum accuracy around 85%). Moreover with a Naive Bayes network only trained on messages, the accuracy starts at 0%, while the user with the network method, initialized with documents, already receives a correct classification in 58% of the cases. Thus, out of the 170 messages, only 72 need to be corrected in the network method, whereas with Naive Bayes all 170 messages need to be labelled.

8.5. DISCUSSION

We have presented a network-based algorithm that has several advantages. First and foremost the presented algorithm is suited for transductive transfer learning since it is capable of using items as training examples that are only partially related to the examples that need to be categorized. The benefit is that using this method already

existing categorizations such as folder structures on the file system can be re-used.

Traditional classification methods like K-nearest neighbour, SVM and Naive Bayes have difficulties in extracting the features from documents that are needed for e-mail categorization(Sappelli, Verberne, and Kraaij, 2012). They can be helped by pre-filtering the documents on a specific type of features, such as contact details. However, these methods can not combine multiple feature types easily. The network-based method presented in this chapter is more flexible. The method is capable of focusing on specific feature types, without the need to filter out other feature types. More importantly, the network approach is not harmed by feature types that are not very effective by themselves, such as the terms, but it can still use them for small improvements.

The classification accuracy reached with the proposed network is much higher than the accuracy of traditional supervised methods on the task of classifying e-mail messages with documents as training data. Using the proposed method, an application can provide reasonable category suggestions at first, and as the user corrects or confirms the system the accuracy can improve to levels almost as good as state of the art for supervised methods that do use labelled e-mail data. A reasonable initial classification substantially lowers the bar to use the system. Additionally it decreases the total amount of effort required from the user, as less training examples need to be labelled. With a reasonable initial classification, many items only need to be confirmed, rather than corrected, requiring less effort as well.

The presented system would be especially meaningful for situations where the effort it costs to label a message outweighs the benefits that the labelled messages has. An example stems from a person that does categorize messages for external projects, but finds it too much effort to label internal projects. The reason for this comes from the nature of the projects; external projects typically have clear boundaries, and little overlap in contacts. Also new people join as the project progresses. Thus, they are easy to label, and labelling them has advantages as it is important to keep track of all the agreements that have been made. Internal projects on the other hand have much higher overlap and are therefore more difficult to label by the user. The presented system would help the user label the external projects by reducing the number of mails that actually need to be labelled, and provides additional benefits to the internal projects that would otherwise not have been labelled. This example also demonstrates that it is not always easy for the user to label data and that help provided by suggested labels by a system could prove beneficial.

A disadvantage of the network is that it cannot discover new categories. However, the network structure allows for the possibility to use graph clustering methods to find clusters of information. This could potentially be used as a method to identify new categories, which we will look into in future work.

Other things that we will be investigating further are the flexibility of usage of the algorithm. First of all, many sources of information such as social information, location information and topic information can be combined in a simple manner. Hierarchically organized information could for example be represented naturally using additional context information levels, where each level in the network corresponds to a level in the hierarchy. Additionally, text features do not have the bulk-advantage

as they have in some algorithms, making it easier to combine them with different feature types. This allows the model to be used in several research domains. Currently we are investigating the benefits and issues when using a more elaborate version of the method for context recognition, where the input data is a stream of sparse events.

Furthermore, the network is very insightful. It is easy to discover relationships between contacts and categories. The method is not limited to categorizations, but can also be used to give insight into a person's data, which is one of the goals in the SWELL project.

The network could be used in a context-aware notification filtering setting as well. This would be a combination of using the model for context recognition and for e-mail categorization. When a new incoming e-mail message fits the current context, a notification can be shown, while if it does not fit, it can be suppressed. This could also be combined with information on the importance of a message (Sappelli, Verberne, and Kraaij, 2013a).

8.6. CONCLUSION

We presented a novel method that is capable of exploiting data from documents as training material for classification algorithms that categorize e-mail messages. The advantage of the proposed method is that the training materials need to be only partially representative of the data that needs to be categorized. This means that more sources of information can be used. Also, categorizations that have already been made by a user in a different, but related domain (such as file organization), can be re-used. In the end this reduces the effort of the user that is typically required when dealing with supervised machine learning systems. We could reduce the number of messages that needed to be labelled or corrected by the user from 170 messages for Naive Bayes to 72 for the network method, in order to obtain a classifier with 80% accuracy in the experiment presented in this chapter. This is a reduction of almost 60%.

9

EVALUATION OF CONTEXT-AWARE RECOMMENDATION SYSTEMS FOR INFORMATION RE-FINDING

Edited from: **Maya Sappelli, Suzan Verberne, Wessel Kraaij** (2016) *Evaluation of context-aware recommendation systems for information re-finding,* Accepted for publication in: Journal of the American Society for Information Science and Technology.

In this chapter we evaluate context-aware recommendation systems for information re-finding that observe knowledge workers in their daily work-tasks. The systems interpret the interaction of users with their regular office PC. The recognized activities are used to recommend information in order to let the user work focused and efficiently.

In order to decide which recommendation method is most beneficial to the knowledge worker, it is important to determine in what way the knowledge worker should be supported. From the knowledge worker scenario we identify four evaluation criteria that are relevant for evaluating the quality of knowledge worker support: context relevance, document relevance, prediction of user action and diversity of the suggestions.

We compare three different context-aware recommendation methods for information re-finding in a task setting where the agent provides the user with document suggestions that support their active (writing) task. Each method uses a different approach to context-awareness. The first method uses contextual pre-filtering in combination with content based recommendation (CBR), the second uses the just-in-time information retrieval paradigm (JITIR) and the third is a novel network-based recommendation system where context is part of the recommendation model (CIA). These methods are also compared to a random baseline.

We found that each method has its own strengths: CBR is strong at context relevance, JITIR captures document relevance well and CIA achieves the best result at predicting

user action. Weaknesses include that CBR depends on a manual source to determine the context and the context query in JITIR can fail when the textual content is not sufficient.

We conclude that to truly support a knowledge worker, all four evaluation criteria are important. In light of that conclusion, we argue that the network-based approach CIA offers the highest robustness and flexibility for context-aware information recommendation.

9.1. INTRODUCTION

Many knowledge workers who process and produce information are confronted with a phenomenon referred to as 'information overload' (Bawden and Robinson, 2009). Knowledge workers can get overwhelmed by the amount of information they need to handle. In our project SWELL[1] we aim to support these knowledge workers in their daily working life by helping them to 'work in context' (Gomez-Perez et al., 2009; Warren, 2013). This means that we try to help the user stay focused on his tasks by recommending documents from the user's work history (documents and webpages that the user has previously accessed) that are relevant to his current working context (i.e. current activities and topics). This chapter describes how we can evaluate context-aware document recommendation systems for knowledge worker support in a document re-finding setting, and provides an evaluation of three approaches to context awareness in document recommendation.

It has been shown that people often forget to use documents that can be helpful, even when they are stored in an appropriate location (Elsweiler, Ruthven, and Jones, 2007). The recommendation of documents can improve task performance by the reduction of the number of computer interactions required, and has been showed to improve the perceived usability of an information re-finding system (Wakeling, Clough, and Sen, 2014). Especially for the task of writing, the time to complete the task can be shortened and the quality of the written document can be improved when relevant information is pro-actively recommended (Melguizo, Bajo, and Gracia Castillo, 2010). This suggests that a recommender system for re-finding information can be useful for a knowledge worker. The time to complete a task or the quality of a written document would be the perfect extrinsic evaluation criteria for the evaluation of a recommender system for re-finding information. However, this data is not available and costly to obtain. Therefore we investigated the potential of using a pre-existing off-line knowledge worker dataset (Sappelli et al., 2014) for the evaluation of simulated context-aware document re-finding.

We argue that there are several ways in which a context-aware recommendation system for information re-finding can be useful to a knowledge worker. For that purpose we describe a knowledge worker scenario. Four evaluation criteria are derived from the knowledge worker scenario, each with their own evaluation metrics. Ideally, a good system would score well on all evaluation criteria. We evaluate three approaches to context-aware information recommendation on each of these criteria. Our research questions are:

[1] http://www.swell-project.net

1. How should we evaluate a context-aware information recommendation system in light of the goal to support knowledge workers in re-finding information?

2. What are the benefits and downsides of content-based recommendation with pre-filtering, just-in-time information retrieval, and context-modelling as methods for recommending documents with the purpose of helping the knowledge worker?

We focus on supporting the knowledge worker through document recommendation, which is why we present a discussion of related work on document recommendation, just-in-time information retrieval and context-aware recommendation in Section 9.3. In Section 9.4 we present the four criteria of evaluation which are derived from the knowledge worker scenario. This is followed by an experiment in which we compare the effectiveness of three methods for incorporating context for recommending documents in a knowledge worker setting. These methods are described in Section 9.5 and the results of the experiment in Section 9.6.

9.2. THE KNOWLEDGE WORKER SCENARIO

A knowledge worker is a person that works mainly with information. He uses and produces information. In our scenario we focus on knowledge workers who work mainly on a computer and process and produce information from documents in order to gain new knowledge.

A typical workday of such a knowledge worker can be described by a combination of activities. Some activities are organizational in nature, such as handling e-mail messages or attending meetings. Some activities are more substantial, such as writing project proposals or reports and preparing presentations. Depending on the type of knowledge worker, software programming or data analysis can also be part of the job.

Consider Bob, he is a 43 year old programmer at a large company. He starts his day with finishing up a report on his latest Java deep-learning project. Only a couple of details and references are needed, but he needs to finish this work before 1 pm. He knows that the papers he needs as references in his report are somewhere on his computer, because he has read them before. At this point he could be helped by opening these documents for him, to spare him the time to navigate to them or look for them himself.

At 11 am he realizes that he is missing a piece of information. He has read it before, but cannot remember where and starts to search on his computer. Bob finds some information about deep-learning in Python, which he also saved on his computer. Because Python is relatively new to him, he finds it more interesting than his current Java project and he gets distracted. At 12.30 he realizes that he has spent to much time learning about deep-learning in Python and that he only has 30 minutes left to finish his project. He finishes it quickly.

In the meantime a couple of e-mail messages have arrived for Bob. Most of them are not so important, but one is about the possibility to work on new, self-defined research. Bob has wanted this for a while, so decides to write a proposal. He already

has an idea about the topic he wants to pursue, but he wants to challenge himself. At this point Bob could be helped by thinking out of the box, and suggesting him documents that are related to the topic, but cover a variety of perspectives.

At 5 pm Bob finishes his day. He has found so many documents for his new project proposal that he feels a little bit overwhelmed. He has not been able to read all documents yet. He decides to catch up on some reading at home.

Our aim is to support Bob in his information management. We see four ways to support him:

(a) By preventing distractions for the knowledge worker so that he can finish his task effectively.

(b) By reminding the knowledge worker of information that he has seen before and is relevant now.

(c) By pre-fetching the documents that he needs for the current task, so that he saves time in navigating to them.

(d) By providing a diverse range of items to spark the knowledge worker's creativity when he needs it.

These four support methods are the foundation for the evaluation criteria that we use to evaluate the context-aware recommendation systems for information re-finding.

9.3. RELATED WORK

In this section we describe previous work related to the research in this chapter. Our work relates to several areas of research: information retrieval, recommender systems (Ricci, Rokach, and Shapira, 2011), information behaviour in context (Ingwersen and Järvelin, 2005) and user-centric evaluation of information systems (Kelly, 2009). In this section we restrict ourselves to related work on a) system-initiated methods for document recommendation (i.e. no search systems) in Sections 9.3.1 and 9.3.3, and b) context-aware methods in Section 9.3.2. In terms of evaluation we focus on off-line evaluation methods, which are described in Section 9.4

9.3.1. DOCUMENT RECOMMENDATION

There are several traditional recommendation approaches to provide users with documents during their work. Most of these make use of collaborative filtering techniques to find relevant documents. Weng and Chang (2008) construct a user profile ontology to reason about the interests of users. They search for user groups with similar interests using a spreading activation model and use their interests as basis for the recommendations of new documents.

In another approach, Lakiotaki, Matsatsinis, and Tsoukias (2011) model the recommendation problem as a decision problem (which document should I use next?), and investigate the use of multiple-criteria decision analysis (MCDA) as method for user profile construction. The authors conclude that MCDA and the subsequent clustering of these profiles enhances the performance collaborative-filtering techniques.

More recently, Lai, Liu, and Lin (2013) have taken the trustworthiness of the ratings by users into account. They propose several methods that use both personal trust as well as group trust. Their proposed methods had lower mean average errors than methods that do not take trustworthiness into account and methods that only use user trustworthiness. This was evaluated on a dataset from a knowledge management system consisting of 800 documents and 80 knowledge workers with their access and rating behaviour.

Although these methods are valuable, they only consider the user and his (relatively static) interests, and not the user's current working context. The recommendations are aimed to be of general interest to the user. These interesting items are not necessarily useful at the time they are recommended and can be a potential source of distraction. Our goal is to reduce the information overload of a user, not to add to it. For this purpose it is important to look at what the user needs, rather than what the user might like. To avoid overload we focus on re-finding information, therefore the user's needs will be to re-find information sources that are relevant for his current activities. Typically, re-finding involves the user issuing queries (Dumais et al., 2003), but in this chapter the focus is on proactive recommendation of documents that the user has seen before.

The task of re-finding information is strongly related to memory (Elsweiler, Ruthven, and Jones, 2007). This has lead to the hypothesis that contextual elements can also help people to re-find items. Blanc-Brude and Scapin (2007) investigated what people recall about documents they have seen and what this implies for search tools. They found that the aspects of documents that users often recall are contextual elements such as keywords, file location, file type, document format, time of last usage, associated events and visual elements. In addition, Kelly et al. (2008) conclude that as the recall of the content itself declines, contextual information becomes more important to re-find information.

More recently, Chen and Jones (2014) have investigated the usefulness of episodic context in a search system for re-finding information. They describe experiments in which they assess the episodic features people remember, which include the name of desktop applications and websites, the name and contact of an e-mail and the information that represent the content of the activity. Although the episodic or contextual features were not frequently used in queries, they did improve the effectiveness of re-finding tasks. File extension, contact names and temporal information were most often used as contextual attributes to a query.

By using the current context of a user, we can find documents that have a relation to a similar context. Since the context of document access, such as the person to which a document was sent or the day it was accessed, can be used to more effectively re-find a document, it is likely that a list of documents related to the user's current context contains documents that the user would potentially want to re-find. This means that for the task of re-finding information we should look at recommendation systems that take context into account. In the next section we describe such recommendation systems that take the user's context into account.

9.3.2. CONTEXT-AWARE RECOMMENDATION AND RE-FINDING SYSTEMS

There are roughly three methods to incorporate context in a recommendation agent: contextual pre-filtering, contextual post-filtering and contextual modelling (Ricci, Rokach, and Shapira, 2011).

In the paradigm of contextual pre-filtering, the set of data that the recommender system uses is filtered for the context that is currently active. This means that simply all the possible suggestions that are not relevant for the current context are taken out before the ranking is determined. Typically the context in these kind of systems is some kind of category. For example a context for a movie recommendation system can be 'watch with family' or 'watch with friends'.

Pre-selection of contexts can be done by using the context as a query. For example, Sappelli, Verberne, and Kraaij (2013c) use the physical location of a user as query and rank the resulting potential tourist activities according to the user's preferences. When this pre-selection is too strict (e.g. there are too few search results for this context), context generalization can be applied (Ricci, Rokach, and Shapira, 2011). In the tourist recommender system, this can be achieved by using a city as location query, rather than the exact GPS location.

A second method for incorporating context in a recommender system is contextual post-filtering. This is very similar to the pre-filtering case, but here the system produces a ranked list for all items, first ignoring any contextual information. The ranked list is re-ranked or filtered afterwards based on the context of interest (Ricci, Rokach, and Shapira, 2011).

There is a last type of context-aware recommendation system where the context is part of the recommendation model. Oku et al. (2006) propose a contextual version of SVM where context-axes are incorporated in the feature space. Incorporating context using factorization methods is also popular (Karatzoglou et al., 2010; Rendle et al., 2011).

The downside of these methods is that the detection of what context is active is often not incorporated in the model. Typically the user is asked to select the context for his search. For example, he can select that he is watching a movie with friends tonight. This means though that all possible contexts need to be determined beforehand, and no personal contexts can be taken into account.

From the perspective of the knowledge worker, his most important context is the (topic of) the task he is working on. As the activities vary throughout the day, it would cost the knowledge worker much effort if he would have to indicate this each time he changes activities. This would diminish the possible advantages of using a recommendation system.

Additionally, reducing the context of a knowledge worker to fixed categories is a limitation, as slight variations in topics would not be captured. A more realistic and content-rich context of a knowledge worker would be the text of a (web) document he is observing at that moment.

9.3.3. JUST-IN-TIME INFORMATION RETRIEVAL

A special type of context-aware recommendation systems are the systems for just-in-time information retrieval (JITIR). In this setting, the context is used as a query in a

search system. The system is pro-active in the sense that the querying takes place in the background, and the search results are presented to the user. Thus, the user does not need to select his context, which is an advantage over the context-aware recommendation systems described in the previous section. The context query that is used can be formulated from the document a person is writing (Budzik and Hammond, 2000; Melguizo, Bajo, and Gracia Castillo, 2010), the blogpost he is writing (Gao and Bridge, 2010), e-mail messages that are being read or composed (Dumais et al., 2004), the news that is being broadcasted (Henzinger et al., 2005) or the text that is visible on screen together with the location, person, date and subject information (Rhodes, 1997).

A limitation of the JITIR systems is that the information leading up to the current context is ignored. The session information can contain valuable information about what has already been seen and what not. Historic behaviour of users has proven to benefit personalized re-ranking of documents (Cai, Liang, and Rijke, 2014).

9.4. EVALUATION FOR CONTEXT-AWARE INFORMATION RECOMMENDATION

Ideally a context-aware information recommendation system for re-finding would be evaluated in an on-line interactive setting with users. In such a case, each user would work as he normally does, while receiving suggestions from one of the systems that is being evaluated. During the experiment we could evaluate whether the suggestions lead to improved task execution in terms of time profit or quality. Moreover, the user could be asked to rate the suggestions he receives at a certain moment. This method of evaluation, however, is expensive. Each system, or even each adaptation in system settings, would require a new period of evaluation with users. Furthermore, the extrinsic evaluation methods are not trivial: to assess time profit or quality of work, the tasks that are being evaluated should be equal. However, if a person executes the same tasks multiple times, there is a learning effect that should not be confused with the effect of using the system. Moreover, asking a user to provide ratings of the suggested documents during the experiment could influence the subsequent suggestions as they are dependent on what is happening on the user's screen, while rating the suggestions afterwards would make the ratings not context-dependent.

To overcome the issues of interactive evaluation, we opt to do an offline evaluation instead. For this purpose we define several criteria that a good context-aware document suggestion should meet. The criteria are motivated by the methods in which we can support knowledge workers as described in the knowledge worker scenario (Section 9.2). We use a dataset of knowledge worker activities (Sappelli et al., 2014) to simulate the work session of a knowledge worker, which enables us to evaluate the recommendations in a context-dependent setting. The assumptions underlying this approach do limit the generalisability of the conclusions. On the other hand, however, this off-line way of evaluation has the advantage that the impact of small changes in system settings can be evaluated more easily. Moreover, it provides the possibility to reproduce results and provides a baseline for comparison for new systems.

There are multiple existing methods for the off-line evaluation of (non context-dependent) recommender systems. Therefore we describe some standard evaluation practices for recommender systems in the remainder of this section. This is followed by the evaluation criteria that we have derived from the knowledge worker scenario (Section 9.2

9.4.1. STANDARD EVALUATION PRACTICES FOR RECOMMENDER SYSTEMS

In the off-line evaluation of recommender systems, the most important measure is predictive-based (Ricci, Rokach, and Shapira, 2011). The assumption is that a system with more accurate predictions of what the user will do will be preferred by the users. There are two interpretations of predictive accuracy in recommender systems. In the first interpretation the system tries to predict the user's rating of an item. Mostly this form of evaluation measures how well a system is capable of predicting how an item will be rated (e.g. movie ratings).

The second interpretation of predictive accuracy focuses on what the user will do with a suggestion. In this interpretation the evaluation focuses on how well a system can predict the action of a user. In a movie recommendation example this would focus not on how the user rates a movie, but on whether the user will actually buy or watch the suggested movie. Both aspects of predictive accuracy are useful to find documents that can support the knowledge worker. For example, we can predict whether a document contains relevant information, or we can predict which document a user will open next.

The case of the knowledge worker is not completely comparable to most recommendation systems. In terms of evaluation, the occurrence of false positives has a larger impact in knowledge worker support than in other recommendation systems such as movie recommendation. In movie recommendation, a bad recommendation will only have a small impact on the overall opinion about the system as long as there are not too many bad recommendations. In the case of the knowledge worker, a bad recommendation can distract the worker and disrupt his work flow, something that is diametrically opposed to the reason for using the recommendation system in the first place. This means that preventing distracting suggestions is an important aspect in the knowledge worker scenario.

We address four possibilities to support knowledge workers in the case of re-finding information, connected to the support options (a)–(d) from the knowledge worker scenario:

- **Context Relevance**: A knowledge worker can be supported by suggesting him documents that fit the topic of his current activities and therefore do not distract him

- **Document Relevance**: A knowledge worker can be supported by suggesting him documents that contain relevant information for the (writing) task he is working on. Where context relevance evaluates whether there is a general topical match with the current activities, document relevance is aimed at a more

detailed evaluation of how much a suggested document contributes to the writing process.

- **Action Prediction**: A knowledge worker can be supported by suggesting him documents that he is going to open in the near future
- **Diversity**: A knowledge worker can be supported by suggesting him a variety of documents

Each of these support possibilities can be seen as a criterion for evaluation. Each evaluation criterion has its own evaluation metric. We have chosen evaluation metrics for each of these criteria based on literature and availability of data. Therefore we start with a description of the data that is available to us, and then describe the evaluation metrics for each criterion.

9.4.2. DATA

For the experiments described in this chapter we make use of a publicly available dataset collected during a knowledge worker experiment (Sappelli et al., 2014). To our knowledge this is the only public dataset with comprehensive computer interaction data that captures the context of knowledge workers realistically and without privacy issues. The interaction data allows for the simulation of a work session, in order to evaluate the context-aware recommendation process.

The dataset was collected during an experiment in which 25 participants were observed while executing typical knowledge worker tasks. The participants were asked to write reports on a total of 6 given topics and prepare presentations for three of the topics. The topics were 'Stress at Work', 'Healthy Living', 'Privacy on the internet', 'Tourist Attractions in Perth', 'Road trip in USA', and 'The life of Napoleon'. So, for each participant we have 6 written documents and 3 presentations that were produced for the task. In addition, we stored all (local and web) documents that were accessed by the users during their work session.

The data were collected in three sessions that mirror the knowledge worker scenario. Each session lasted between 30 and 45 minutes. The conditions were: a) a neutral session in which the participants were asked to work as they normally do; b) a session in which they were time pressured and c) a session in which they were interrupted with email messages. Some of these messages contained a task for the participant, which resulted in two additional topics in the data: 'Einstein' and 'Information Overload'.

The dataset that resulted from this experiment contains all computer interaction data that was recorded during the experiment. Most importantly the dataset contains the data originating from the uLog key logger[2] as well as browser history data collected with IEHistoryView.

For the experiments described in this chapter we make use of the preprocessed version of the dataset. In this version of the dataset, individual events are aggregated to meaningful *event blocks*. The start of a new event block is defined as either an application switch event, or a change in window title event. All the individual

[2] http://www.noldus.com/human-behaviour-research/products/ulog

events, such as the keys typed, and all captions (mouse-over tool tips), that occurred between application or window switches are concatenated into strings and the number of mouse clicks per event block is counted. From the recorded Google URLs the queries that were entered were extracted using a regular expression. In total, the data collection consists of 9416 event blocks with an average of 377 event blocks per participant. The average duration of an event block is 51.5 seconds. The average number of accessed documents per participant per 3-hour work session was 44 documents.

Table 9.1: Overview of features collected per event block, with example values

feature	example value
id	6
participant id	2
begin time	20120919T132206577
end time	20120919T132229650
duration (seconds)	24
# clicks	3
typed keys	we austra;i lia
application	iexplore
window title	Google - Windows Internet Explorer
caption	New Tab (Ctrl+T) New Tab (Ctrl+T)
url	http://www.google.nl/search?hl=nl& sclient=psy-ab&q=australia+&oq=australi
domain	www.google.nl
query	australia
Label	Perth

Table 9.1 shows an overview of the features collected per event block, with an example value for each feature.

DATA LABELLING

Table 9.1 shows that each event block was labelled with a topic label. This was done as a second step after the data collection experiment using the crowd sourcing platform Amazon Mechanical Turk. The event blocks were presented to the annotators in a desktop-like setting to mimic the desktop view of the user during the experiment. The annotators were asked to select 1 topic label and also indicate on a scale of 1-5 how certain they were of their decision. The event blocks were shown in random order, so they could not use any session information. The labels were the 8 topics ('Stress at Work', 'Healthy Living', etc.), and an additional topic 'indeterminable' when the event block did not contain any identifiable topic, for example when just the website 'www.google.nl' was shown.

Each document that was opened during the experiment was labelled with the topic label that was assigned to the event-block in which the document was accessed. A document can have multiple topic labels. In total there were 799 documents accessed during the experiment, of which 349 were tagged with the label 'indeterminable'. We assume that within one event-block, a single topic guided the information access behaviour of the user. Table 9.2 presents the distribution of doc-

uments over the topic labels. Overall there were 43 documents that were associated with more than one topic. An example is *http://healthypattern.com/things-you-can-do-at-work-to-relieve-stress-at-work.html* which is associated with the topics Healthy Living and Stress. Some of these documents have multiple labels because of errors in the labelling by Amazon Mechanical Turk. An example of such a document is *http://www.perthtourism.com.au/recreation.html* which is associated with both Roadtrip and Perth. The roadtrip topic was about a roadtrip in the USA, so a website on Perth should not have been tagged with this label. We have not corrected these erroneous labels, as this kind of noise would occur in a live system as well.

Table 9.2: Overview of documents per topic

Topic	Number of documents
Einstein	19
Privacy	30
Information Overload	13
Roadtrip	138
Perth	127
Healthy Living	70
Stress	58
Napoleon	88

USING THE DATA FOR RECOMMENDATION

For the evaluation of context-aware re-finding we assume that the user is writing a document or preparing a presentation, similar as in this dataset. For the simulation of the re-finding task, we need a set of documents that the user has accessed before (in reality maybe weeks or months earlier), either stored locally on his computer or visited in his browser. The set of documents with a label other than 'indeterminable' that are accessed during the experiments is on average 44 documents per participant, which is too small to evaluate a typical knowledge worker setting. Therefore, we extended the list of candidate documents with the documents accessed by all users. This is a set of 450 documents of which 95% are web-documents defined by an URL. The dataset shows that on average a knowledge workers accesses 18 documents per hour, thus we argue that the set of 450 documents represents a history of at least 25 hours of concentrated work. In reality this would be equivalent to a working week, since a normal working day also includes other activities such as meetings etc. We argue that the set of 450 documents is large enough to introduce re-finding problems. For each participant the work session is simulated by re-running the logged event blocks. For each event-block we determine the relevancy of each of the documents in the collection, rank them and select the top 10 as our recommendation list. This is motivated by the length of a typical search result page (10 search results). However, the optimal number of suggestions in context-aware document recommendation is an open topic for research that is not within the scope of this chapter.

Documents that are open in the current event-block or have been opened in previous event-blocks in the current work session are filtered. The assumption is that

documents that have been seen in the current work session should not be recommended because the user does not need help re-finding those. We assume that a session consists of the activities that are executed between system boot and system shut down, with a maximum duration of one day. The expectation is that documents that are accessed during a day are remembered by the user and do not need to be recommended. In the dataset the session of a participant equals the three-hour experiment in which he participated.

The recommendation lists are evaluated on the four knowledge worker support possibilities: context relevance, document relevance, action prediction and diversity. It is possible that providing recommendations for each event-block is too often. This should be optimized in future work. The reason that we choose to provide recommendations for each event block is that this represents the dynamic nature of the context well. In the next subsections we will describe an evaluation criterion with an evaluation metric for each of the support possibilities.

9.4.3. CONTEXT RELEVANCE

A first possible criterion in the evaluation of a context-aware recommendation system involves the evaluation of whether the suggested documents fit the user's current context. We aim to help the user focus on his activities, so suggestions that are related to a different context would possibly distract the user. In this evaluation measure, we define a correct context as a topical match between a suggested document and the current activities.

For this evaluation criterion we use the topic labels that are assigned to each document. These topic labels can be seen as a category of 'context' and are equal to the topic-labels of the event-blocks. If the category of a suggested document matches the category of the current activities (e.g. the current event-block), we consider the suggestion to be a good one. We assess the quality of the recommendations using precision@10 (how many recommendations in the top 10 have the correct context).

9.4.4. DOCUMENT RELEVANCE

Although a topical match to the active context is interesting, it does not mean that a document that is suggested can be used by the knowledge worker. For example a knowledge worker producing a manual for some software will use different sources than when he is writing a report on the project for which the software was produced even though the context is the same. Therefore we consider the criterion of document relevance, which evaluates how relevant a suggested document is for the specific task the knowledge worker is working on.

Ordinary document relevance can be assessed by obtaining relevance judgements. However, for context-aware systems document relevance judgements need to be obtained within the context that the document was accessed. This means that we would need a document relevance assessment for each document in each context, and for each user separately. These relevance judgements are hard to obtain and are not available in the dataset.

An alternative to using manual relevance judgements is to look at the dwell time for each document. The advantage is that these are measured within context, so if a

document is accessed within multiple contexts, multiple dwell times are measured. We investigated the appropriateness of dwell time in the dataset as criterion for relevance. If we use a threshold of 30 seconds (Guo and Agichtein, 2012), then only 44 documents in our data set would be estimated as relevant. This is only 1.3% of all documents in the dataset, which seems unrealistically low. One explanation comes from copy-paste behaviour. Some users tend to quickly copy some text from a viewed document to the document they are producing. This makes the dwell time for the viewed document low, even though the copy-behaviour suggests that the document is highly relevant. Also, when users quickly switch between the viewed document and the document they are producing, the individual dwell times are low.

Recent work has shown the limitations of using a (single) dwell-time threshold as relevance indicator and other evaluation metrics should be taken into consideration (Lehmann et al., 2013; Kim et al., 2014). In the dataset we use (described in Section 9.4.2) there was a strong focus on the production of texts. When we interpret document relevance as those documents that contain text that is used by the participants, we can use textual overlap between a suggested document and a produced document as an indicator for relevance of the document. The assumption is that the more relevant a document is, the more similar it will be to the produced document. This captures copy-paste behaviour that we observed in the data as well, since there will be a high similarity when complete sentences or paragraphs of one document occur in the other. Using this approach we can obtain personalized context-aware document relevance scores for each participant. The limitation of this measure is that a produced document needs to be available in order to determine the relevance.

For this purpose we use the ROUGE-N measure by Lin (2004). This measure is originally intended for the evaluation of summaries or translations. It uses the number of overlapping n-grams between a source and a target document and is defined by:

$$score = \frac{2 * \left|source \cap target\right|}{|source| + \left|target\right|} \tag{9.1}$$

where we use the set of word 2-grams in the recommendation as source and the set of word 2-grams in the written document by the participant as target. In our interpretation a high ROUGE-score means that the document that is considered had a high contribution to the document that was produced by the user. In the original version of the measure, the score is normalized on the length of the user-produced document. However, as there can be a large difference between the length of the user-produced document and the candidate document, we normalize the score on the length of the sum of the documents. This length normalization is performed after stop-word removal [3].

Each produced document by a participant was tagged with the corresponding task context in which it was produced (the context labels of the event blocks). There were typically 6 produced documents per participant, one for each of the main tasks in the data collection. There were no produced documents for the tasks 'Einstein' and 'Information Overload'.

[3] Stopwords retrieved from `http://snowball.tartarus.org/algorithms/english/stop.txt`

For each candidate recommendation, the ROUGE score is calculated between the candidate document and the produced document of the participant that was tagged with the label of the active context (i.e. the label of the event-block for which the recommendations are generated). In the case of webdocuments, html tags are removed before the ROUGE score is calculated. When there was no produced document available (i.e. 'Einstein' and 'Information Overload' or if a participant had not produced a document for the task), then the document relevance was automatically 0.0.

We assessed the validity of ROUGE-N as measure for document relevance on a randomly selected subset of 80 documents in their context. Two human assessors were shown two documents at a time: the produced document by the participant (the context), and the document to assess.

They were asked to provide a rating on a 5-point scale on how relevant the assessment document was for the creation of the produced document. There was a significant positive correlation between the ratings and ROUGE-N (Kendall's $\tau = 0.663$, $p < 0.001$), which means that a higher ROUGE-N score is associated with a higher human rating. Furthermore there was a substantial inter-annotator agreement on 20 overlapping items (weighted Cohen's $\kappa = 0.68$ (Cohen, 1968)). The positive correlation indicates that ROUGE-N can be used as measure for document relevance.

9.4.5. ACTION PREDICTION

With the third evaluation criterion we evaluate the known item recommendation as an action prediction problem; which document will the user access next? If we can predict this document, and would present it to the user, this would save him the time to locate the document. We evaluate this by looking at the document the user accesses in the next event block. Since not all suggestions lists contain the document that will be accessed next, we consider success@1 and success@10: does the top-1 or top-10 list of recommendations contain the document that will be opened next?

9.4.6. DIVERSITY

With the fourth evaluation criterion, we evaluate how original a document suggestion, or a list of suggestions is. This is in part contradictory to the relevancy criteria, since a diverse set of recommendations is more likely to contain distracting documents. However, we think that diversity is important in order to engage the user with the system. With a large enough document set, diversity should be possible without loosing relevance.

We evaluate diversity by looking at two aspects: uniqueness of elements and variation between suggestion lists. Uniqueness is motivated as follows: if a recommender system offers more unique recommendations in one event block compared to the surrounding event blocks it is more original to the user than when it provides the same recommendations over and over again. For this aspect we consider how many unique items are recommended in all event blocks with the same context (a measure of catalog coverage (Ricci, Rokach, and Shapira, 2011)) . This is measured with:

$$score = \frac{1}{|C|} \sum_{x \in C} \frac{unique_x}{total_x} \tag{9.2}$$

for a context $x \in C$, where $unique_x$ is the number of unique documents that occur as suggestion for a context, and $total_x$ is the total number of documents that have occurred as suggestions for a context.

The second aspect is variation between suggestion lists. If subsequent suggestion lists are highly similar (e.g. the same suggestions in the same order), regardless of the actual activities of the user, the suggestions may not impact the user. Then the user will not consider the new suggestion list as original and he will not look at it. For this aspect we consider Rank Biased Overlap (RBO) as measure for rank correlation (Webber, Moffat, and Zobel, 2010). RBO measures the similarity in ordering between two lists and is calculated using:

$$score = (1-p)\sum_{d=1}^{n} p^{d-1} A_d \tag{9.3}$$

where d is the position in the list, n is the size of the list and A_d is the proportion of the two lists that overlap at position d . The parameter $p = 0.9$ models the user's persistence (will a user look at the next item in the list). This gives more importance to the top ranked items than to the lower ranked items. This measure has the benefit that it is not hindered when there is no or little overlap between the top 10 results (compared to other rank correlation measures such as Kendall τ). If there is no overlap than the RBO score is 0.

9.5. METHOD

In this section we describe three different approaches to context-aware information recommendation. Sections 9.5.1, 9.5.2 and 9.5.3 describe the three approaches and their implementation with the used dataset. Their effectiveness is evaluated and discussed in Section 9.6.

9.5.1. JITIR SYSTEM

We implemented a just-in-time IR system as follows: For each user all 450 candidate recommendation documents were first indexed using the Indri Search Engine[4]. We used the Indri API to set up a query interface. For each event block in the data a query was constructed. This query consisted of the typed keys, window title and the text from the url or document that was active. All characters that are not alphanumeric, no hyphen or white space are removed from the query terms. As ranking model we used the default Indri Retrieval Model. The top 10 results, or less when there were less than 10 results, were considered for evaluation.

The JITIR system is hypothesized to perform well on document relevance, as it is has a focus on finding documents that contain terms that have been recorded in the current context as well.

[4]http://www.lemurproject.org/indri/

9.5.2. CONTENT-BASED RECOMMENDATION WITH CONTEXTUAL PRE-FILTERING

We implemented a content-based recommendation system (CBR) with pre-filtering as means to incorporate context-awareness. This type of system is dependent on a (manual) categorization of the active context and the candidate documents in order to filter the candidate documents.

The dataset provides manually assigned context labels for each event block in the data. These labels correspond to the topics from the knowledge worker tasks(e.g. 'Napoleon', 'Healthy living'). Each document was assigned one or more context labels based on the labels of the event blocks in which the document was open. During run-time, the subset of documents with the same context as the event-block was selected. Then the items in the subset were ranked based on their cosine-similarity to the document that was open in the event-block. The features that were used were the normalized TFIDF scores on all terms in the documents. Documents that were more similar to the open document were assumed to be more relevant. When there was no document open in the event-block, the documents with the correct context were ordered at random.

We hypothesize that the active filtering of items with the wrong context has a positive effect on the performance on the context relevance criterion.

9.5.3. CONTEXT-AWARE RECOMMENDATION WITH CONTEXT DETECTION

The context-aware recommendation system with context modelling for context detection that we implemented is a novel method based on the interactive activation model by McClelland and Rumelhart (1981) and depicted in Figure 9.1. The added benefit of this method compared to CBR is that it does not depend on a manual source to determine what context is presently active. Compared to JITIR it has the benefit that it takes the history into account using decay.

In essence an advantage of the CIA network approach is that it could function as a memory extension for the user: The network stores explicit associations between information entities, similar as how the user would associate items. The idea of nodes, connections and spreading activation has relations to the working of the brain (Anderson and Bower, 1973). This could potentially benefit the recommendation, as it can use similar contextual cues for recommending items as a person would have used.

The network consists of three main layers:

- the document layer: this layer contains nodes for all 450 candidate recommendation documents This corresponds to the access history of approximately 25 hours (assuming on average 18 documents per hour)
- the context information layer: this layer contains nodes for the context information, divided into four categories of context information types: terms or topics, entities, locations and date/time elements
- the event layer: this layer is the input for the network. Here the sensed/recorded

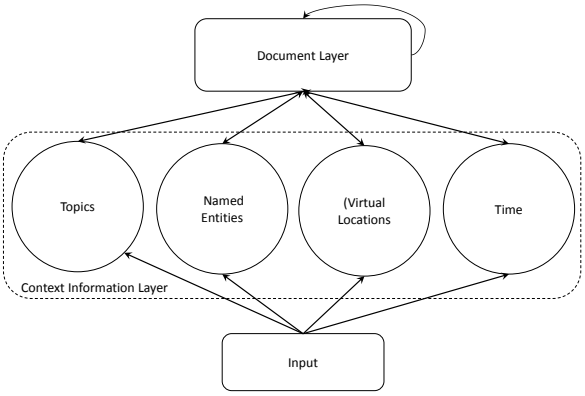


Figure 9.1: The Contextual IA model (CIA). The figure shows the 3 layers. There are activating bidirectional links between the context information layer and document layer and unidirectional links from input to context. There are no links within layers, except for activating links between documents that are based on temporal closeness between opening documents

event-blocks enter and activate the network. In our dataset an event-block is a collection of events (key activity, mouse activity, window title, url) that was recorded within one tab or window of an application. This means that the event blocks are variable in their duration.

In the network the event layer activates the context information layer by observing which terms, entities, virtual locations and time information are present in the recorded event block. In its turn the activated context information nodes activate documents that are described by this context information; for example a term node 'health' activates all the documents that contain the word 'health'. Then the activated documents enhance the activity in the network by activating all their context nodes, for example the document that was just activated because it contained the word 'health' now activates the word 'well-being' as well because that word was also present in the activated document.

This spreading activation method serves as a sort of pseudo relevance feedback, mitigating the sparseness of the information in the event blocks. However, due to the sparseness of incoming information, there is a risk that too much irrelevant information is activated in the document layer. To prevent this 'snowball effect' of sparseness, we implemented a TFIDF-like weighting for the connection weights: The connection weights from context information to documents are based on inverse node frequency and the connection weights from document nodes to context information nodes are based on node frequency. As a result, observed context information that occurs in many documents has less impact than information that occurs in only one document. And information that occurs frequently in a document has a bigger impact than information that only occurs once in the document. There are only positive (excitatory) connections in the network. A detailed motivation for the choice of connection weights can be found in table 9.3.

Table 9.3: Connection strengths between the various node types. These are the weights on the activation flow from one node to another. They are based on the concept tf-idf term weighting. $\#outlinks$ refers to the number of outgoing connections of a node.

From	To	Value or function	Motivation
Event-block	Date/Time	1.0	An event has one unique time stamp
	Entity	$\frac{\#entity_x \in event}{\#entities}$	Strength of activation of an entity should be dependent on how strong the entity is present in the event, proportional to the number of all entities in the event
	Location	1.0	An event has at most 1 location
	Topic	$\frac{topic_x \in event}{topic_{1..n}}$	Strength of activation of a topic should be dependent on how strong the topic is present in the event, proportional to the number of all topics in the event
Date/Time	Document	$\frac{1}{\#outlinks}$	Multiple documents can be accessed on the same date, or hour.
Entity	Document	$\frac{1}{\#outlinks}$	entities that occur in many documents should be less influential
Location	Document	1.0	
Topic	Document	$\frac{1}{\#outlinks}$	topics that occur in many documents should be less influential
Document	Date/Time	1.0	
	Entity	$\frac{\#entity_x \in document}{\#entities}$	Strength of activation of an entity should be dependent on how strong the entity is present in the document, proportional to the number of all entities in the document
	Location	1.0	A document only has one location
	Topic	$\frac{topic_x \in document}{topic_{1..n}}$	Strength of activation of a topic should be dependent on how strong the topic is present in the document, proportional to the number of all topics in the document
Document	Document	$\frac{1}{\#outlinks}$	A document can have a relation to multiple other documents

For each event-block the network is activated for 10 iterations. The difference in activation from one iteration to the next is defined using Grossberg's activation function:

$$\delta a = (max - a)e - (a - min)i - decay(a - rest) \tag{9.4}$$

where a is the current activation of a node, e is the excitatory input of the node, i is the inhibitory input and min, max, $rest$ and $decay$ are general parameters in the model. The excitatory input pushes the activation to the maximum, while the inhibitory input drives it down to the minimum. The decay parameter gradually forces the activation back to its resting level when there is no evidence for the node and allows for cross-over of network activation from one event-block to the next. For pragmatic reasons, the network is not run until convergence, but only for 10 iterations. This is enough for sufficient activation in the network. The assumption is that the documents with the highest activations after those 10 iterations are the best candidates for suggestion.

In this chapter we compare 2 variations of the CIA approach that vary in the method that is used to determine which topics or terms are representative for the context of interest. In the first approach, CIA-t, we use the top 1000 terms from the term extraction method described in Verberne, Sappelli, and Kraaij (2013) as representative terms. These 1000 terms are also extracted from events and documents.

In the second approach, CIA-lda, we use a latent dirichlet allocation model (LDA model) instead of term extraction to model the topics. LDA is often used for topic extraction. In this setting we have used the MALLET implementation of LDA (McCallum (2002)) and 50 topics are extracted. The initial LDA model is trained for 1500 cycles on a set of manually selected Wikipedia pages (e.g. the Wikipedia page 'Napoleon' for the topic Napoleon), one for each of the tasks from the experiment. The same topics are also extracted from events and documents. For both CIA-t and CIA-lda we use the Stanford Entity Recognizer trained for English (Finkel, Grenager, and Manning, 2005) to determine which entities occur in event blocks or documents . The values of the parameters are the same as in the original IA network: $min = -0.2$, $max = 1.0$, $rest = -0.1$, $decay = 0.1$

The CIA system in general is expected to perform well on diversity as it incorporates a form of query expansion, which allows for unexpected suggestions. This will be a trade-off with context relevance and document relevance, as more original suggestions will have a higher risk of being less relevant.

Another criterion on which CIA is expected to perform well is the prediction of which document a user is going to open. This is because CIA incorporates direct associations between documents, based on previous document access as well as document content.

Since the evaluation metric for document relevance is based on term overlap, we expect that CIA-t has an advantage over CIA-lda on the document relevance criterion as CIA-t also has a focus on terms rather than topics.

9.6. RESULTS

For the discussion of the results, this section is divided into the four subsections that correspond to the four evaluation criteria described in Section 9.4. We compare the three methods described in the previous section to a baseline recommender system that randomly selects 10 documents to suggest from the list of all 450 candidate documents. Documents that are open or have been opened before in the same session were excluded from the list of candidate documents. All significance values reported in this section are based on a paired samples t-test with a 95% confidence interval. The results are the macro averages over the event-blocks. Thus, first the average per participant is calculated. Then the average of these averages is reported to ensure that each participant has an equal effect on the average, regardless of the number of event-blocks in his session. The macro averaging method provides an estimate of the simulated system performance on each evaluation criterion averaged across 25 participants, using recorded standardized task guided – but natural – interaction data for approximately 3 hours (including short breaks).

The CIA-lda method uses an LDA model for topic recognition. Since LDA is non-deterministic, there could potentially be a difference in results between different initializations of the LDA model. Therefore, the reported results of CIA-lda are averaged over 5 runs, with 5 different LDA models. The differences between runs are not significant: $p = 1.000$.

9.6.1. CONTEXT RELEVANCE

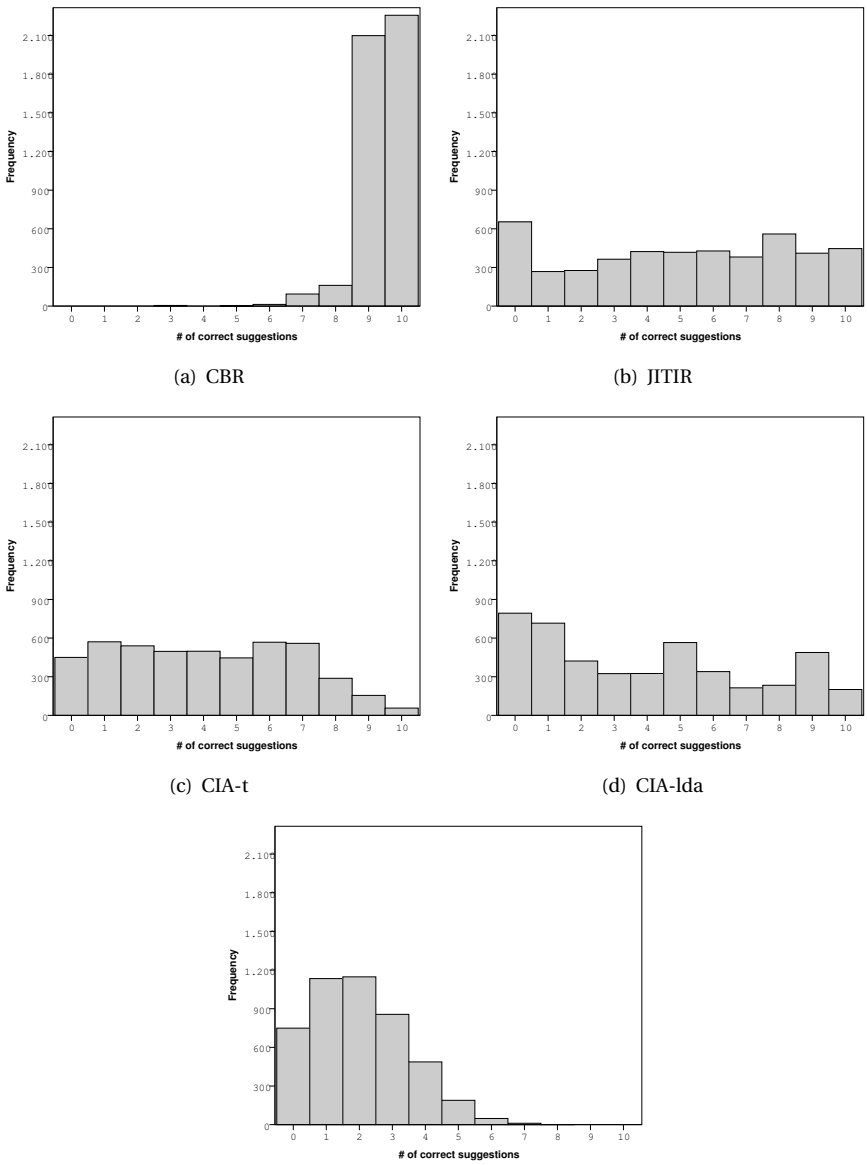
Table 9.4: Accuracy of the context of the suggestion.

Measure	CBR	JITIR	CIA-t	CIA-lda	Random
Precision@1	**97.7%**	59.1%	36.0%	44.2%	20.7%
Precision@10	**94.1%**	50.0%	39.7%	40.0%	19.6%

Table 9.4 shows the results for the match of the recommendation to the context. In addition, we present histograms for each recommendation method that show how often, how many of the 10 suggestions have the right context (Figure 9.2).

The table shows that the CBR approach is most effective in finding suggestions that match topically to the context (e.g. where the label of the document matches the label of the event-block). This is trivial as the CBR uses a hard filter on the context. Nevertheless, the histogram in Figure 9.2(a) that there are also event blocks for which CBR cannot provide 10 correct suggestions. In those cases, CBR does not have enough candidate documents remaining for the context after filtering the documents that have already been opened in the session. This happens in 51.3% of the event blocks.

The JITIR system has a top document suggestion with the same topic as the active context in 59.1% of the cases. When the entire list of 10 suggestions is evaluated,



(a) CBR

(b) JITIR

(c) CIA-t

(d) CIA-lda

(e) Random Recommendations

Figure 9.2: Histograms of the context relevance of suggestions

5 out of 10 suggestions have the right context on average. Both results are significantly lower than CBR ($p < 0.001$). Both the CIA-t network, and the CIA-lda networks have significantly lower success rates for its top recommendations (36% and 44.2% respectively, $p < 0.001$). CIA-t and CIA-lda suggest approximately 4 out of 10 suggestions with the right context, which is significantly lower than CBR and JITIR ($p < 0.001$). The JITIR system, however, cannot suggest any documents in 3.9% of the event blocks, because of query-failure. CIA can always suggest the requested amount of documents, provided that there are sufficient candidate documents.

The random approach shows that on average 2 out of 10 suggestions will have the correct context when picked randomly, which is significantly lower than CBR, JITIR, CIA-t and CIA-lda ($p < 0.001$). The histograms in Figure 9.2 show that both CIA and JITIR show more uniform distributions over the number of correct suggestions, while CBR has a clear peak at 9 and 10 correct suggestions. The random system typically has 0, 1 or 2 correct suggestions within its suggestion list.

Since the CIA approach attempts to classify the context at the same time as it recommends documents, it is possible that there is a relation between the number of suggested documents with the right context and whether the context was accurately predicted. A one-way anova revealed that indeed the average number of suggestions with the correct context is significantly higher when the correct context was predicted ($p < 0.001$). For CIA-t 4.8 out of 10 suggestions had the correct context in the case of a correct prediction, while in case of a wrong prediction 2.8 suggestions were correct. For CIA-lda the difference was slightly smaller: 4.5 out of 10 for correct predictions, and 2.9 for wrong predictions.

9.6.2. DOCUMENT RELEVANCE

Table 9.5: Relevancy of the suggestion lists measured with ROUGE-N. (max) denotes the score for the best suggestion in the list, while (avg) denotes the averaged score for the entire list

Measure	CBR	JITIR	CIA-t	CIA-lda	Random
to written (max)	0.0149	**0.0172**	0.0117	0.0083	0.0053
to written (avg)	0.0031	**0.0049**	0.0023	0.0020	0.0005

Table 9.5 shows that regardless whether the complete suggestion list, or the best item in the list was considered, the recommendations by JITIR were most valuable (avg = 0.0049 and max = 0.0172). These values are significantly better than CBR, CIA-t, CIA-lda and random ($p < 0.001$). A score of max = 0.0172 indicates that the textual overlap between the best candidate in the list and the produced document is 1.7% on average (over event blocks and participants). In this dataset, the maximum relevance that can be obtained for the best candidate document in a suggestion list for an event block is 0.6830. This is the ROUGE-score for a candidate–context–participant combination where the participant copied a large part of the candidate document in a particular task context. However, generally the scores are much lower: 84% of the candidate–context–participant combinations have a ROUGE score of 0%.

When the performance of the other systems is considered, CBR suggests more relevant documents (max and avg) than CIA-t, CIA-lda and random ($p < 0.001$). Moreover CIA-t suggested more relevant documents than CIA-lda ($p < 0.001$), which is what we expected, considering that CIA-t has a stronger focus on term overlap because of its term extraction method. This suggests that methods that have a strong focus on term matching such as JITIR have an a priori advantage on this metric.

Finally, both CIA-t and CIA-lda suggested more relevant documents than the random system ($p < 0.001$). The random system has an especially low performance if the entire list is considered (avg=0.0005), which is close to the average ROUGE score of 0.0007 for all candidate–context–participant combinations.

9.6.3. ACTION PREDICTION

Table 9.6: Predictive power of user's action.

Measure	CBR	JITIR	CIA-t	CIA-lda	Random
Success@1	0.0041	0.0053	0.0197	**0.0253**	0.0001
Success@10	0.0163	0.0148	0.0477	**0.0483**	0.0014

Table 9.6 shows that CIA-t and CIA-lda have better predictive power than JITIR and CBR. The success@10 measure reveals this; CIA-t and CIA-lda will predict the next document correctly in its list of suggestions in 4.8% of the cases (the difference between them is not significant: $p = 0.564$). JITIR only predicts the next document correctly in 1.5% ($p < 0.001$). CBR predicts the next document with 1.6% predictive accuracy, which is comparable to JITIR ($p = 0.502$) and worse than CIA ($p < 0.001$). When the top suggestion is considered (success@1) CIA-lda performs better than CIA-t ($p = 0.008$). Both CIA-t and CIA-lda are better than CBR and JITIR ($p < 0.001$) and JITIR is better than CBR ($p = 0.023$). All systems are better than random ($p < 0.001$).

Note that these predictive accuracies are rather low. This is a side effect of the requirement in the systems that they cannot recommend documents that are opened in the session already. Since the key log data includes frequent switches back and forth between documents, many of the recurrent openings of documents cannot be predicted. The theoretical maximum average predictive power is 42.6%, since 67.4% of document access events are of the type re-opening during the session.

Even though CIA, and specifically CIA-lda, performs the best on action prediction, CIA is potentially harmed by the manner of document selection. In the document selection, documents accessed by all participants are included in the list of candidate documents. However, for most of these documents, the simulated access data of the participant is not available. This means that the time nodes and document to document connections that CIA would normally create during on-line learning of interaction have not been created for the simulated experiment. These type of connections are especially relevant for predicting which document is going

to be opened.

Table 9.7 shows the predictive power of the various methods, if the access pattern would have been available. This data is based on an experiment where only the documents accessed by a participant are included as candidate documents (on average 44 candidate documents). The table shows that the success rates of all methods improve because of the reduction in number of documents. CIA benefits the most: the success@10 of both CIA-t and CIA-lda increase to 12%, while the theoretical maximum predictive accuracy remains 42.6%. Interestingly in this case the difference between CIA-t and CIA-lda on Success@1 is not significant anymore ($p = 0.629$)

Table 9.7: Predictive power of user's action, personal document set.

Measure	CBR	JITIR	CIA-t	CIA-lda	Random
Success@1	0.0245	0.0170	**0.0503**	0.0498	0.0025
Success@10	0.0636	0.0570	**0.1247**	0.1238	0.0165

9.6.4. DIVERSITY

Table 9.8: Variability in the suggestion list. Rank Biased Overlap (RBO) is measured with $p = 0.9$. RBO was measured for a suggestion list compared to all other lists in the session, as well as a suggestion list compared to the suggestion list of the next event. A low RBO value represents a larger diversity

Measure	CBR	JITIR	CIA-t	CIA-lda	Random
RBO - Session	0.468	0.195	0.245	0.331	**0.059**
RBO - Next Event	0.465	0.137	0.135	0.238	**0.059**
Unique Suggestions	12.6%	12.5%	14.2%	12.8%	**17.9%**

Table 9.8 shows that the suggestions by the random system have the highest variability (this means lowest RBO) in their orderings, which is expected given the current definition of diversity. CBR on the other hand shows a high RBO for both the session and the next event (47% commonality between lists). This can be explained by the filtering on context that CBR uses, which limits the choice in candidate documents per event-block. CIA-t, CIA-lda and JITIR score a bit in between in terms of variability. For session variability there is a 33% commonality (RBO) between recommendation lists for CIA-lda, 25% for CIA-t and 20% for JITIR, whereas the commonality from one event to the next is 24% for CIA-lda, and 14% for both CIA-t and JITIR.

In terms of unique suggestions, the random baseline has the highest number of unique documents in its suggestion lists, followed by CIA-t. The difference between the number of unique suggestions for CIA-lda, CBR and JITIR is minimal. Overall,

CIA-t scores slightly better on this criterion than CIA-lda, JITIR and CBR because of more unique suggestions in combination with a low RBO score between events.

9.7. DISCUSSION

We have presented four evaluation criteria that are relevant for the evaluation of knowledge worker support in the task of information re-finding. In Section 9.7.1 we start with a discussion of the evaluation measures to answer the question "How should we evaluate a context-aware information recommendation system in light of the goal to support knowledge workers in re-finding information?" (RQ1)

In Section 9.7.2 we continue with a discussion of three context aware recommendation approaches and their performance on the four evaluation criteria. This answers the question what the benefits and downsides are for the various approaches for recommending documents with the purpose of helping the knowledge worker. (RQ2)

We conclude with a discussion on the limitations of this work and some suggestions for future work in Section 9.7.3.

9.7.1. EVALUATION CRITERIA AND METRICS

The evaluation criteria that were described in this chapter cover several aspects of knowledge worker support. Some of these evaluation criteria may be related. For example, if a document is relevant for the user, it is not likely that this document will distract the user (i.e. does not match the active context of the user). Therefore, we measured the correlations between the metrics that were used to assess the four evaluation criteria.

A two-tailed Pearson correlation test reveals that context relevance is moderately positively correlated with average document relevance ($\rho = 0.445$, $p < 0.001$). This means that indeed a document that does not fit the current activities is not likely to be relevant.

The other measures have negligible correlations. Action prediction is negligibly correlated with context relevance ($\rho = 0.040$, $p < 0.001$) and document relevance ($\rho = 0.062$, $p < 0.001$). Diversity as measured with rank biased overlap (RBO) is negligibly uncorrelated with document relevance ($\rho = -0.008$, $p < 0.001$) and action prediction ($\rho = -0.018$, $p = 0.187$), but weakly positively correlated with context relevance ($\rho = 0.122$, $p < 0.001$).

These correlations suggests that the document relevancy measure might be redundant. We should, however, in the future look at a situation where there are multiple writing tasks with similar topics, to fully understand the document relevancy measure. Nevertheless, since some of the context-aware recommendation methods are focused on using context categories, while others use a more elaborate context, it seems reasonable to evaluate both tasks separately. Otherwise there might be a bias towards the type of context-aware approach already.

Moreover, there are still aspects of the ROUGE-metric for document relevancy that need to be considered. In this chapter we have used stop-word removal, length normalisation and the removal of html tags as preprocessing steps for the ROUGE-

metric. An aspect that we have not considered is the selection of text that needs to be considered for the metric. For example, wikipedia indicates the part of the page that is being watched with a suburl (i.e. `https://en.wikipedia.org/wiki/Napoleon_Bonaparte#Early_career`). However, the text that is used for the calculation of document relevance is based on the entire webpage, since the webcrawler extracts the entire page, not just the part that is being watched. In general, the crawled webpages are a source of noise. Sometimes the actual text cannot be extracted, for instance when the page is in Flash.

Another point for discussion is the definition of the diversity measure. At this point diversity is measured independent of relevance. However, recommending diverse but irrelevant documents is not beneficial for the knowledge worker. This shows that it is important to consider the various evaluation criteria in combination. By measuring them in isolation, an incomplete picture about the performance of a system is sketched, which becomes apparent when we consider the performance of the random system on the diversity measure.

Overall, when we consider the knowledge worker and his situation as a whole as described in the scenario we prefer a method that scores well on all evaluation criteria. After all, a system that can prevent distractions really well (context relevance) is not useful when it only suggests the same documents over and over again (diversity). A system that can predict which documents will be opened is not useful when these documents will distract the user.

9.7.2. CONTEXT-AWARE RECOMMENDATION APPROACHES

When we consider the performance of the various context-aware recommendation approaches on the four evaluation criteria, we can conclude that there is no single recommendation method that yields the best results on all dimensions. It depends on the task at hand, what the best recommendation method is to support the knowledge worker. This means that, considering the variety of activities that the knowledge worker is involved during a day, the best recommendation method can vary even in a single day of work. Therefore, it is important to continue to work towards a context-aware recommendation approach that scores well on all tasks and is not dependent on explicit human context assignments.

If the goal of the system is to prevent distractions for the knowledge worker (context relevance), the content-based recommender system with contextual prefiltering (CBR) shows the best results. This supports the hypothesis that CBR is good at preventing distractions because it actively filters documents with the wrong context. This result is expected, and illustrates why it is important to consider multiple evaluation criteria. Also note, that although a context match implies that the document is no distraction, a document with the wrong context does not need to be a distraction if it provides relevant information for the task (e.g. a document that is tagged with 'Stress' could also be relevant for the active context 'Healthy Living').

If the goal of the system is to suggest documents that are likely to contain relevant information that the knowledge worker can use, then JITIR is the best choice, both when the complete suggestion list is considered as well as when only the best item in the list is considered. For this criterion, systems which suggest documents that

textually overlap with the current context have a benefit.

If the goal of the system is to predict which documents a knowledge worker will open, then CIA is the best choice, especially when the document access patterns are available (which is the default case, since CIA has been designed to take advantage of interaction patterns). This supports our hypothesis that CIA has an advantage in action prediction because there are direct associations between documents based on the time-of-opening in the CIA approach. Moreover CIA provides top results in document relevancy.

If the goal of the system is to provide a high diversity in results (regardless of relevance), then the random system should be used. This is a result that could be expected given the current definition of diversity. CIA shows promise in terms of diversity as well, especially when term extraction is used (CIA-t). CIA-t suggests more unique documents than CIA-lda, JITIR and CBR. Of course, the fact that the diversity measure does not take relevance into account is a limitation of the measure. Overall CIA, JITIR and CBR are preferable over the random system as they will recommend more relevant documents by design.

Regardless of their performance on the evaluation criteria, each recommendation method has advantages and disadvantages. CBR has the advantage that it is a simple and robust method. However, CBR is sensitive to a cold-start problem that occurs for every new context that is introduced. If there are no or few documents that are tagged with the active context, than CBR cannot provide a sufficient amount of recommendations. Because of the hard filter, CBR cannot use documents that are tagged with a different but strongly related context, even though these might be good suggestions. Furthermore, CBR depends on a manual source to determine which active context is currently active, which requires more user effort. This is especially the case in the knowledge worker scenario, where the context is highly dynamic.

The advantage of JITIR is that it does not depend on an external source for context determination. The use of context as query is simple and effective, and there is no need for context categorization. The downside is that sometimes this query fails, so that no recommendations can be provided. This occurs in 3.9% of the event blocks.

In essence an advantage of the CIA network approach is that it could function as a memory extension for the user: The network stores explicit associations between information entities, similar as how the user would associate items. With this mechanism it is a step towards the design principles formulated by Elsweiler, Ruthven, and Jones (2007) to improve personal information management systems. The disadvantage of CIA, is that its recommendation lists have a lower context accuracy. However, the flexibility of the method can be used to improve performance on certain criteria such as document relevance, for instance by using term extraction instead of topic modelling.

When we consider the complete knowledge worker situation as described in the scenario, we judge CIA as the most promising approach of the three. CIA is good at predicting which document the user will access next and provides a diverse set of recommendations. Although its recommendation list might contain documents that do not strictly match the current context, overall it seems to contain at least one good suggestion for most event blocks.

9.7.3. LIMITATIONS AND FUTURE WORK

One aspect of evaluation that is lacking is the real-time performance and scalability of the approaches. This is an aspect that is important when a system is put to practice, especially for context-aware systems. A system is not likely to be useful to the knowledge worker when the suggestions are not provided in time. Since our dataset contained not enough data to pose problems for scalability and did not contain data over multiple days, we have not considered these dimensions in this chapter.

A further limitation of the research presented in this chapter is that there was only one dataset that we could use. Its characteristics may explain some of the generally low performances on document relevancy. The document data in the set was not filtered for noise and contained data in at least two languages. Additionally the dataset contained no actual relevance judgements, causing us to divert to derivative measures. For a proper evaluation of the methods and evaluation metrics, we should look at the performance on a second dataset.

In future research it is important to investigate what users value most in context-aware recommendation systems. How often should the system recommend documents, and how many documents should be in the suggestion list? Another important aspect is the further exploration of the evaluation metrics that we have used. Are these the optimal ones, or are there alternatives that have a stronger relation to the user's preferences? Although we have presented an alternative to dwell-time and document relevance judgements in the form of ROUGE-N, its characteristics need to be explored further to see the potential of the measure as alternative measure for document relevance. Moreover, the diversity metrics should be adapted to take the relevance of the suggestions into account in order to make the criterion less trivial.

Furthermore we propose to consider a task-dependent cost-based metric in the future to determine which recommendation strategy to use at a certain time. The cost should be dependent on the characteristics of the task the knowledge worker is executing. This would allow the design of a hybrid context aware recommendation system that can optimally support the knowledge worker in various circumstances. For example it could stimulate diversity when the knowledge worker is exploring a new topic, while focusing on context relevance when the knowledge worker needs to finish a task.

9.8. CONCLUSION

In this chapter we have described the evaluation of context-aware document recommendation with the purpose of supporting knowledge workers in a re-finding setting. The scenario of the knowledge worker is different from typical context-aware recommendation scenario's as the context is more dynamic and there is larger negative impact of irrelevant recommendations. In this chapter we have presented and used a dataset that facilitates research to this kind of complex recommendation scenario's. We focus on four evaluation criteria that are relevant for knowledge worker support: context relevance, predicting document relevancy, predicting user actions and diversity of the recommendation lists.

We have evaluated three different approaches to context-aware document recommendation in a realistic knowledge worker setting where the context is given by

the interaction of users with their regular office PC. One approach to context-aware document recommendation is a content-based recommender with contextual pre-filtering (CBR), one is a just-in-time information retrieval system (JITIR) and one is a novel method that is capable of detecting the active context simultaneously to providing context-aware document suggestions (CIA).

The conclusion of which context-aware document recommendation method performs best highly depends on the evaluation criterion that is considered. Overall, each method performed well for at least one evaluation criterion. CBR was best at context relevance, JITIR was best at providing a recommendations that are likely to contain text that the knowledge worker will use and CIA was best at predicting which document the user will open. The random baseline was best at providing diversity in its suggestions.

Overall we believe that the CIA approach is most promising for context-aware information recommendation in a re-finding setting as it performed best in terms of action prediction, while providing diverse results as well. Moreover, CIA is not dependent on human effort for detection of the active context. Nevertheless, there is room for improvement when it comes to document and context relevance. The flexibility of the system provides ample opportunities to investigate these aspects. Finally, we conclude that the multi-faceted evaluation approach allows for a more comprehensive view on the quality of context-aware recommendation systems for supporting knowledge workers.

10

RECOMMENDING PERSONALIZED TOURISTIC SIGHTS USING GOOGLE PLACES

Edited from: **Maya Sappelli, Suzan Verberne, Wessel Kraaij** (2013) *Recommending personalized touristic Sights using Google Places*, Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2013), Dublin (poster).

The purpose of the Contextual Suggestion track, an evaluation task at the TREC 2012 conference, is to suggest personalized tourist activities to an individual, given a certain location and time. In our content-based approach, we collected initial recommendations using the location context as search query in Google Places. We first ranked the recommendations based on their textual similarity to the user profiles. In order to improve the ranking of popular sights, we combined the initial ranking with rankings based on Google Search, popularity and categories. Finally, we performed filtering based on the temporal context. Overall, our system performed well above average and median, and outperformed the baseline — Google Places only — run.

10.1. INTRODUCTION

According to a report from the The Second Strategic Workshop on Information Retrieval in Lorne (submitted to SIGIR Forum, 2012), "Future information retrieval systems must anticipate to user needs and respond with information appropriate to the current context without the user having to enter an explicit query". At TREC 2012, a new track was organized: the contextual suggestion track[1], in order to evaluate such proactive systems. In this track the goal was to suggest personalized tourist activities to an individual, given a certain geo-temporal context.

[1] https://sites.google.com/site/treccontext/

As input to the task, each group participating in the track was provided with a set of 34 profiles, 49 examples of tourist activities and 50 geo-temporal contexts in XML format.

Each tourist activity example consisted of a title and a short description of the activity as well as an associated URL. The tourist activity examples were a collection of bars, museums and other tourist activities in the Toronto area.

Each profile corresponded to a single user and consisted of a list of rated URLs of the tourist activity examples in Toronto. The ratings were divided into an initial rating, based on the title and description of the URL and a final rating, which was given by the user after he/she viewed the website. These ratings could be used as training data to infer the particular preferences for this user.

For testing, systems needed to generate suggestions for 50 geo-temporal contexts. Each context consisted of spatial information (city-name, state-name, latitude, longitude) and categorical temporal information (day, time and season). The day could be a weekday or a weekend day and the time was either morning, afternoon or evening.

The task for the participating teams was to build a system that automatically provides a ranked list of 50 suggestions for each profile/context pair. Each suggestion should contain a title, description and associated URL. The description of the item could be personalized. The suggestions should be appropriate to the profile as well as the geo-temporal context. Time-wise, the user has five hours available for the suggestion, limiting acceptable locations of suggestions.

For evaluation, a selection of these suggestions were rated by the persons that provided the profiles, and the suggestions were assessed on their fit to the spatial and the temporal context by professional assessors as well.

Although there is quite some research in the area of mobile tourist guides, only a few works describe automatic recommendation of tourist places based on interests and context. Ardissono et al. (2003) describe their Intrigue system which presents a user with tourist information in the Turin, Italy region. They define heterogeneous tourist groups (such as families with children) and recommendations are made while taking possibly conflicting preferences into account. Preferences are given by the users themselves and reflect geographic features, essential information such as opening hours, basic information such as price, specific characteristics such as the historical period of an attraction, and properties such as historical value. In a conflicting group the preferences of individuals are weighted and compared to the properties of an activity to determine its rank.

Schwinger et al. (2005) do not present a ready to use system, but study the strengths and weaknesses of several mobile tourist guides. They note that current systems tend to use their own selection of content data. This gives the developer more control over the presented information, but it also means that rich tourist-content websites are not used. Some systems adapt to the user's interests, but they require the user to provide these interests or at least explicit feedback on the points of interest.

Buriano (2006) shares his views on the importance of social context in tourist activities. He notes that people enjoy sightseeing in groups and that they involve

their social networks by sharing pictures for example. He suggests that these social relations should be included in recommender systems for tourist activities.

These works suggest that it would be wise to exploit the expertise of specialized websites. Also automatic personalization is an interesting approach, with the note that the social context should play a role as well.

In the contextual suggestion track, however, the user profiles were anonymous. We did not have any demographic information of the user, or information about the user's social situation. This limited our options. Therefore we have taken a content-based recommendation approach. We selected potential tourist activities from Google Places using the context information and re-rank these potential places to match the user's preferences. In section 10.2 we describe our recommendation approach. The results were evaluated in several ways, which is described in section 10.3, after which we finish with a discussion in section 10.4.

10.2. METHOD

Our method comprises 5 steps: (1) Collecting a first set of potential recommendations, (2) building the user profiles, (3) ranking the recommendations for the user profile, (4) re-ranking the list of recommendations, (5) filtering the recommendations using the temporal context. A more detailed description of these steps can be found in Sappelli, Verberne, and Kraaij (2013b);

(1) Collecting potential recommendations

The first step was to collect potential recommendations for tourist places. We used the Google Places API for that purpose. Longitude and latitude of the location were used together with the keyword "tourist attractions" to retrieve relevant places. Short descriptions of the search results were obtained by querying the Google Custom Search API with the URL of the search result from Google Places.

(2) Building the Profiles

We described a user with two term profiles, one with terms of tourist examples judged positively by the user and one with terms of examples judged negatively. Terms from the title and description from the examples where put in the positive term profile if the *initial* rating was positive, and in the negative if the *initial* rating was negative. Terms from the categories, reviews and events from Google Places where put in the positive profile when the *final* rating was positive or in the negative if the *final* rating was negative. Terms with a neutral association were ignored. We did not use the content of a website, because the websites contained either too much noise (e.g. advertisement data) or we could not extract the content easily (flash content). Overall, this collection of terms results in the user profile $U = \{R_p, R_n\}$ in which R_p is the term frequency vector representation of the "positive" profile and R_n of the "negative" term profiles.

(3) Ranking recommendations

To rank the potential recommendations based on the user models we used two different methods: a similarity based method and a language modelling method.

In the similarity method, each term in the term profiles was weighted using the tf-idf measure (Salton and Buckley, 1988) to determine the importance of each term in the profile.

We represented the potential tourist sight by a tf-idf term vector as well, based on its title, description, reviews and events. The fit of this potential recommendation was determined by taking the cosine similarity between the potential suggestion and the positive and negative profiles. The suggestions are ranked on their similarity scores. We order each items descending on their $cos_{positive}$ score. However, when $cos_{negative} > cos_{positive}$ we place the item at the bottom of the list (i.e. after the item with the lowest $cos_positive$ score, but with $cos_{positive} > cos_{negative}$). Originally, we discarded the items with a better fit to the negative profile than to the positive profile, but we needed them to be able to meet the number of requested recommendations (50 recommendations per person/context combination).

The alternative method we used to rank the potential recommendations was using a language modelling approach. In this variant the Kullback-Leibler divergence was used to weigh each term. We used point-wise Kullback-Leibler divergence (Kullback and Leibler, 1951), as suggested by Carpineto et al. (2001). It functions as a measure of term importance that indicates how important the term is to distinguish the "positive" examples from all examples.

A potential recommendation is better when it has many terms that are important in the "positive" examples. For each potential recommendation we derived its score by taking the sum of the Kullback-Leibler scores for the terms describing the search result. The potential recommendations were ordered descendingly on their scores. This approach benefits suggestions with more textual data, since the likelihood that it contains terms that also occur in the profiles is larger.

(4) Re-ranking the list of recommendations

During the development phase, we had no evaluation material. Therefore, we had to evaluate our methods manually. We created our own personal profile and we looked at which order of suggested activities appealed more to us.

We noticed in the suggestions given by the two runs, that famous tourist attractions did not rank very well. This is likely to be an artefact of the example data. For example, the Statue of Liberty does not resemble any of the examples in the tourist activity examples in Toronto, so it is no surprise that it does not receive a high rank. However, we believe that these famous sites should rank well. Therefore we use elements from the Google Places API to increase the rank of these items, independently of the user profiles.

We take an approach in which we created 4 ordered ranked lists: (A) Our personalized ranking based on KL-divergence or tf-idf; (B) a ranking based on the prominence of a place given by the original order of Google Places; (C) a ranking based on ratings of people that visited the place as indication of the overall perceived quality of a place; and (D) a ranking based on the a priori category likelihood. This latter ranking is based on the idea that some people have preferences for certain categories of activities (such as museums) rather than preferences for individual items. We derived the ranking from the Google categories and the times that this category appeared in positive and negative examples. This final rating was smoothed (using +1 smoothing) to account for categories that did not occur in the example set. Since these were quite a lot and we did not want this to influence the results too much we weighted this rank half as much.

The final rank is determined by the weighted average rank of the search result in these 4 ordered lists. The weights we used were $\{1, 1, 1, 0.5\}$

(5) Filtering based on temporal context

In the last phase, we filter out the search results that do not match the temporal part of the given context using manually defined rules. We use the opening hours as registered in Google Places as reference material for determining whether a result matches the temporal context or not. For example, when the temporal context is evening, we do not suggest search results that have opening hours until 5pm.

(6) Presentation of the results

The first impression of a search result is very important for its relevance assessment by the user. However, some Google snippets contained advertisements or unclear descriptions. Therefore, we decided to use positive reviews as descriptions for the suggested places. Even though they might not always be good descriptors for the suggestion we hope that the positiveness may make people more inclined to give a positive rating.

10.3. RESULTS

In this section we present the accuracy and precision@5 results that we obtained with the two runs we submitted: (1) run01TI ranking based on tf-idf with cosine similarity and (2) run02K ranking based on point-wise Kullback-Leibler divergence scores. There were only 44 out of 1750 profile/context pairs taken into account during evaluation (i.e. not all contexts, and not all profiles were evaluated) and only the top 5 suggestions were evaluated. All results in this section are based on these 220 (i.e. $44 * 5$) data-points.

Table 10.1: Precision @5 results for both runs and the –Google Places only– baseline

	Website*GeoTemporal	Description	Website
run01TI	0.19	0.42	0.40
run02K	0.22	0.41	0.47
baseline	0.18	0.30	0.41
	Geotemporal	Geo	Temporal
run01TI	0.54	0.89	0.56
run02K	0.57	0.90	0.58
baseline	0.51	0.79	0.57

Table 10.1 shows the precision results for the different measures, as well as a baseline (baselineA) provided by TREC, which is based on the original order of Google Places. To calculate precision, only items that have scored a rating of 2 (i.e good fit, or interesting) on each dimension are considered relevant. The results show that the differences between the tf-idf measure and the Kullback-leibler divergence measure are very small. Both measures seem to perform better than the baseline. Interestingly the geographical fit of this baseline is lower, which is likely caused by a different query method. The results of our runs show a particularly high precision at rank 5 for the geographical fit. The precision in terms of the rating on description and website shows room for improvement. Also the precision on the combination of personal rat-

ings (e.g. website) and geo-temporal fit is not very high. However, the neutral items are interpreted as bad suggestions, making this measure quite conservative.

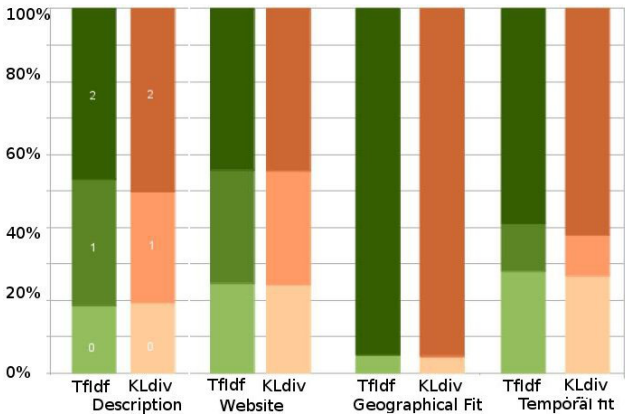


Figure 10.1: Distribution of positive (2), neutral (1) and negative (0) ratings

A more detailed look on the distribution of positive, neutral and negative ratings is given in Figure 10.1. The two left-most columns of Figure 10.1 show that approximately half of the suggestions are perceived as interesting (rating 2) when it comes to the opinion of the users. Many items (a third) are perceived as neutral (rating 1). This may mean that the user is not yet sure if he/she would want to follow up on the suggestion, in any case the user is not negative on the suggestion. Overall, around 80% (the sum of the 1 and 2 ratings) of the suggestions are perceived as positive when only the description is shown. When the website is shown the users are a little less positive.

The two right-most columns of Figure 10.1 show a big difference between the accuracy of the suggestions in terms of the geographical fit to the context and the temporal fit to the context. The difference between the tf-idf measure and the Kullback-leibler divergence measure is again neglectable. 95% of the suggestions fit the geographical context.

The temporal context is matched in 62% of the suggestions. This leaves room for improvement. After inspection we noticed that theatres and night clubs tend to be suggested during the day as well. This is caused by the opening hours of the box office, which are usually in the afternoon and thus according to our algorithm a suitable suggestion for the afternoon context.

10.3.1. IMPACT OF MIXING RANK-METHODS

The impact of each of the ranking methods on the final ranking was assessed using Kendall's τ (Kendall, 1938).

Table 10.2 shows the average rank correlations (Kendall's τ) with the final ranking for the various ranking methods from section 10.2. Overall we see that the rankings based on the user profiles (by either KL-divergence or tf-idf) are correlated the

Table 10.2: correlations between the ranking methods (A,B,C,D) and the final ranking (Kendall's τ)

	with Final Ranking
(A) Tf-idf	0.59
(A) KL-divergence	0.56
(C) Ratings from other people	0.36
(D) A-priori category likelihood	0.20
(B) Place Prominence	0.17

most with the final ranking. The prominence of a place (based on the original Google Places order) has the least influence on the final ranking.

The tf-idf measure and the Kullback-Leibler measure show a correlation with each other of $\tau = 0.47$, showing that the methods are actually quite similar in the proposed order of suggestions, even though the actual ranks may vary. Also both methods are slightly correlated with rankings based on ratings from other people ($\tau = 0.17$ for KL-divergence and $\tau = 0.21$ for tf-idf).

10.4. DISCUSSION

We encountered a number of challenges in the implementation of our approach. First, it was difficult to obtain 50 suggestions for each context. This was mainly because of the limitations of the Google Places API. However, since only the top 5 suggestions were evaluated this did not have an effect on our results.

A second problem was the little variation between suggestions for one person and the other. This was a result of a high similarity between user profiles, which was caused by the limited example set. Each individual rated the same example places and they tended to be very positive about them as well. The rating may be positively biased, since the training examples were places from the area of residence of the users. It is possible that when rating places that you are familiar with, you have other preferences than when it comes to places that you have not visited before.

In general, it is still a point for debate how much the influence of personal characteristics should be when suggesting tourist sights. After all, people often go to the main points of interest when they visit a city anyway. It is important that these are part of the suggestions. But, when a person visits the place for a second time, personal characteristics might be more important, since the person has likely visited the main points of interests already. For some types of suggestions, e.g. places to eat, personal characteristics are likely to be more important than for other types of suggestions. This would be an interesting point for future research.

Most other teams used a similar method for collecting search results. Some groups included more specialized search engines such as Yelp. Many teams used a recommender system based approach in which search results were collected first, and ranked according to their match to term profiles, although a few teams took an approach in which a query was generated based on the user's preferences. Some teams used the terms from examples, others focused more on conceptualizing examples by recognizing categories from them.

There were two teams in the top 5 results using tf-idf weighting with cosine sim-

ilarity to calculate the match between profile and search results, while our tf-idf run was at position 11. These two teams did not mix their results with other rankings like we did, used different descriptions and also had a slightly different approach in acquiring search results. Our runs both performed better than average and median and even had the best performance for a few of the contexts.

10.5. CONCLUSION

We think we have several strong points in our approach. Overall it is attractive that our approach is completely automated. Our suggested places matched the geographical contexts very well. This is because we used search results from Google Places, which allowed us to use precise location information in the search query. However, even though opening hours were provided by Google Places as well, it was more difficult to obtain a good fit on the temporal context, because these hours were sometimes erroneous but also because not everybody had the same interpretation of the categorical values of the temporal context.

Secondly, we think it is attractive to mix several ranking methods. This way we could find a balance between personalized suggestions and more generic famous places suggestion. Additionally, we could use the opinion of people that have visited the sight already. Our analysis of the rank correlations for the ranking methods show that the personalized ranking method (either by tf-idf or KL-divergence) had the most impact on the final ranking. Interestingly, both the tf-idf measure and KL-divergence measure rankings correlated slightly with rankings based on the ratings from other people. This means that a personal measure gives to some extent the same ranking order as a collective measure based on ratings by many people.

And finally, we think the use of reviews as a description for the search result is attractive, since it gives a personal touch to the suggestion even though the descriptions are not personalized. A positive review may influence people, making them more enthusiastic about the suggestion. Overall, people responded a little better to our descriptions than to the website (see Table 10.1).

We could make some improvements by investigating the influence of the keyword that is used to collect potential places. Additionally, the weights of the 4 ranking methods could be optimized, once there is more data available.

More generally speaking, the TREC contextual suggestion track provides a platform to evaluate the "zero query term problem" in which the search engine can proactively suggest resources given a context. In the future this can be expanded with context types, other than geo-temporal context, such as social context, or content context.

11

REFLECTION AND CONCLUSION

In this thesis we investigated the use of context-aware algorithms to support knowledge workers in their battle against information overload. In the first part of the thesis we improved the knowledge about the behaviour of knowledge workers during their work in data collection experiments. In the second part we defined the context that is necessary to support a knowledge worker and how we can recognize context automatically from the interactions of the knowledge worker with his computer. Finally, in part 3 we described context-aware and personal information management methods that require little user effort and which support knowledge workers . We will first discuss the research questions for each part of the thesis before we discuss the main research question: *"How can we design, implement and evaluate context-aware methods that make computer-based knowledge work more effective and more efficient?"*. We conclude with some suggestions for future work.

PART 1: UNDERSTANDING THE KNOWLEDGE WORKER

Improving the well-being of knowledge workers by means of context-aware applications requires a thorough understanding of what the knowledge worker needs. In other words, we need an understanding of a knowledge worker's intentions when he is working to determine how we should support him. In this first part of the thesis we collected data to get a better understanding of the knowledge worker's intent in several work-related activities. In this section we will summarize the findings of Chapters 2, 3 and 4, followed by a discussion on the first research question: *"What information about knowledge worker intent can we observe from interactions with the computer and what information do we need to deduce from other sources?"* . We start with an overview of our research contributions.

In this part, we present three datasets as main contributions. The annotated dataset on query intent is especially relevant for the Information Science and Information Retrieval community, as it presents the query interpretation from the searcher himself.

The second dataset with annotated e-mail messages provides a multi-dimensional classification of the messages with a focus on the tasks specified in the messages. The dataset opens research into a) understand how tasks are conveyed in e-mail messages (Information Science), and b) evaluate new e-mail categorizations for the support of knowledge workers (Information Retrieval).

The third dataset with logged computer interactions of a knowledge worker during typical knowledge worker tasks includes heart rate, skin conductance and facial expressions. This makes it a dataset that can be used in many fields of research. The Information Science community can use it to understand how people interact with a computer and how they feel during knowledge work. Additionally, the Information Retrieval and Recommender System communities can use the dataset to investigate context-aware algorithms for knowledge work support, with a new range of context information that can be used.

CHAPTER 2: COLLECTING GROUND TRUTH DATA FOR QUERY INTENT

Searching for information is one of the activities that a knowledge worker will engage in when working. In Chapter 2 we investigated the knowledge worker's intent when he is engaging in search activities. We do so by logging the queries that he issued during work. In query intent research, the logged queries are typically annotated by independent assessors. In our work, however, we asked the searchers themselves to annotate their own queries. We used a multi-dimensional annotation scheme that provides insight in what the knowledge worker was hoping to achieve by issuing the query. The scheme includes dimensions on the general topic of the intent, the type of action, what kind of information the knowledge worker wants to retrieve (modus), how detailed the searched information should be (specificity), how reliable the source of the information should be (source authority sensitivity), and whether the requested information is sensitive to a certain location or time (location and time sensitivity).

One of the goals of the study was to improve automated query intent classification. As automated systems only have access to the textual content of queries and not the original intent, we analysed the terms in the query and their relation to the annotated form. From this data we concluded that the textual content of the queries does not give many hints as to what annotation in terms of modus, action, source authority sensitivity, location sensitivity, time sensitivity and specificity can be expected. Also, the length of the query did not predict the specificity of the query intent, meaning that even when the query intent was very specific, the query could just as well consist of only one or two terms. We suggest that taking into account contextual sources of information, such as the current computer interactions or the search history could provide a better understanding of the query intent.

CHAPTER 3: ASSESSING E-MAIL INTENT AND TASKS IN E-MAIL MESSAGES

Another activity that knowledge workers engage in frequently is e-mail communication. We believe that e-mail is a medium in which knowledge workers often communicate about the work tasks that they are involved in. In Chapter 3 we investigated how knowledge workers use e-mail messages to communicate about the tasks they

are involved in. We first discussed the reliability and validity of assessing e-mail intent in a pilot study. Furthermore we annotated messages from two e-mail datasets with e-mail intent and task intent when there was a task communicated in the message. We developed a novel multi-dimensional annotation scheme that gave insight in the e-mail intent in terms of the e-mail act (type of message), the expected response, the reason for sending the message, and the number of tasks for the recipient in the message. In terms of task intent, we looked at the time and location sensitivity of the task and the task type of the task. For the annotation of selected messages from the public dataset (Enron) we made use of crowd-sourcing through Amazon Mechanical Turk, while we used expert annotators for the selected messages from the licensed dataset (Avocado).

From this data collection we can conclude that most messages are sent to deliver information or to request information. Requests are not often rejected. Only half of the messages require a reply, but this reply does not have to be immediate. When we look at the tasks in the messages, approximately half of the e-mail messages contains a task for the recipient. Typically not more than one task is communicated through the message. Most of these tasks can be executed everywhere (low spatial sensitivity). Some tasks do have a high or very high time sensitivity such as a deadline, but the likeliness of this happening depends strongly on the company. The evolution of a conversation reveals that there is a high probability that a deliver message is followed after another deliver message. This suggests that much information is delivered, even without a request. Furthermore, requests are followed by a message that delivers the requested information, or a message in which the recipient commits to the request. A message that delivered information is often followed by a "greet" message, which was most likely a thank you message.

CHAPTER 4: COLLECTING A DATASET OF INFORMATION BEHAVIOUR IN CONTEXT

A final source of information about knowledge worker intent can be derived from the interaction of the user with the computer. In Chapter 4 we described the collection of a dataset of such computer interactions. This human-computer interaction data consists of data collected from a key logger such as keystrokes, active applications,etc. and a browser-logging application. Because this type of data is very privacy-sensitive this data was collected in a controlled experiment that mimicked typical knowledge worker work. In the experiment, 25 participants were asked to write reports and prepare presentations on several topics. The experiment consisted of three conditions; one neutral baseline, one where the participants were pressured for time, and one where the participant was interrupted with e-mail messages that could contain an additional task.

Initial analysis of the interaction data showed some challenges in working with this type of rich data. First, combining information from multiple sources proved challenging as some information was not logged at all or was logged multiple times. Also, logging browser URLs proved to be challenging as query suggestions by Google could result in incomplete URLs that were logged. Furthermore, the dataset contained much noise caused by on-page advertisements, plug ins and icons.

The data in this collection gave insight into activities outside the browser, but also into information seeking behaviour. This behaviour was natural, as it was not

explicitly part of the original assignment (write a report or prepare a presentation). An interesting finding was that even though page dwell times were low, these pages could still be relevant for the task. Logged dwell times were lower than expected because of copy-paste activities, or switching between applications.

RQ 1. WHAT INFORMATION ABOUT KNOWLEDGE WORKER INTENT CAN WE OBSERVE FROM INTERACTIONS WITH THE COMPUTER AND WHAT INFORMATION DO WE NEED TO DEDUCE FROM OTHER SOURCES?

From the data collections collected and described in this thesis we can conclude that much information about the knowledge worker's intent is implicit. Interactions in the form of queries only reveal part of the original intent of the knowledge worker. Understanding the intent of an e-mail message and the embedded tasks are easier to understand than a query. Still, some information remains implicit. Part of this implicit information could be captured to some extent by looking at contextual information; what is happening outside of a query or a message. Computer interactions such as copy-paste behaviour reveal important information about what is used by the knowledge worker to complete his task, which can be used to interpret the original intent. The challenge in using computer interactions, however, is that there are so many of them that it is difficult to integrate them properly, and that they contain much noise.

Overall we can conclude that a high-level topical understanding of the knowledge worker's intent of e-mail messages and queries is deducible from their textual content. However, for a more detailed understanding, such as the spatial and time sensitivity of a query we need other sources of information.

Concerning e-mail messages, we need to distinguish between the intent of the sender and the tasks for the recipients. Both are valuable sources of information that can give information about the knowledge worker. If the knowledge worker is the recipient of a message, typically half of the messages contain a task for him. The textual content of the e-mail messages reveals a sufficient amount of information to interpret the spatial and time sensitivity of the task as well as the general type of task (informational, physical or procedural). If the knowledge worker is the sender of the message, then the textual contents can be used to understand whether the user expects a response, and what the implicit reason for sending the message was (e.g. collaboration, administrative procedure etc.).

In addition, computer interaction data provide insight into the relevance of data sources by means of which documents are accessed, how long they are observed, but also whether text is copied from the source.

Limitations The limitation of our research is that we only investigated queries, e-mail messages and general computer interactions. There are many more sources of information that could be used to interpret the knowledge worker's activities and his intent such as calendars and task lists. Another possibility is to look at other resources that the knowledge worker uses, such as his phone or the people he interacts with. Finally, in our computer interaction analysis we have focused on text that was visible and text that was used. We realize that the document selection behaviour,

the order of access, or deletion behaviour reveal important aspects of relevance of information as well. This means that there are many aspects that still need to be researched to fully understand which information about knowledge workers we can observe from interactions, and which information needs to be deduced from other sources.

Additionally there are some limitations to the datasets that we have used. The limitation of the query intent dataset that we have collected is that the participants are all from a Computer Science background, which limits the generalizability. The limitation of the e-mail dataset is that the messages are collected before 2005. In order to generalize our claims, additional data from 2005-2015 should be investigated. And finally the limitation of the knowledge worker dataset is that it was collected during a controlled experiment. Although we tried to make the setting as realistic as possible, the participants may have been influenced by the tasks they were given, and the fact that they were observed using cameras and physical sensors. Moreover, the data was collected in 3 consecutive hours per participant, meaning that the data does not contain long term behaviour.

PART 2: CONTEXT OF THE KNOWLEDGE WORKER

We have learned that in order to support a knowledge worker we need to have an understanding of his intentions. In order to do so, we need to capture his context; e.g. what is he doing and what is happening around him. In part 2 of the thesis we have described what this context could look like and how we can capture it in a way that is conform our assumptions, with little user effort and can be used in context-aware applications.

In this section we will summarize the findings of Chapters 5 and 6 and we discuss the research questions: *"How should we define the context that we need for context-aware support for knowledge workers?"* and *"How can our conceptual model be implemented and how well can it detect the active context?"*. We start with an overview of the contributions of this part.

There are two important contributions in part 2 of the thesis. The first is a conceptual and formal model of the context of a knowledge worker. The formal model allows for reasoning about the knowledge worker, such as which resources he is going to interact with, which knowledge he is going to learn and what his tasks may be.

The second contribution is a novel algorithm for context recognition and identification that can be used for context-aware support as well. It is founded by the model of context for a knowledge worker, instead of designed from an application perspective. This makes the algorithm applicable to multiple scenarios. The algorithm is relevant for the fields of Information Retrieval and Recommender Systems because of its applicability for context-aware recommendation systems.

CHAPTER 5: THE KNOWLEDGE WORKER AND HIS CONTEXT

In Chapter 5 we described the context of a knowledge worker. We started this chapter with a literature overview on the concept of 'context' as this concept is a possible source for miscommunication. Additionally, we clarified the various positions

on context that have been taken by other researchers in the domain of personal information management. For this purpose we described three scales of context interpretation (container vs. meaning, objective vs. subjective, representational vs. interactional) and positioned the relevant literature on these dimensions.

We continued with a conceptual description of the contextual elements that we consider to be relevant in the area of personal information management. Additionally we presented a formal description of this conceptual model and described how this model can be used to reason about a knowledge worker's actions.

RQ 2. HOW SHOULD WE DEFINE THE CONTEXT THAT WE NEED FOR CONTEXT-AWARE SUPPORT FOR KNOWLEDGE WORKERS?

In our opinion the context of a knowledge worker is highly dynamic and driven by events. We assume that the knowledge worker is the centre of the context, but the context can influence the knowledge worker as well. We assume that we can observe the context using sensors, independent from the knowledge worker, but know that the interpretation of the sensed elements is dependent on the knowledge worker.

In the conceptual model of the context of a knowledge worker we focus on the interaction of the knowledge worker with his surroundings. Both the knowledge worker as well as the resources he interacts with are partly observable and partly unobservable.

The formalisation of the model allows for the reasoning about resource selection and knowledge gain for the knowledge worker. More importantly, the formal model shows how we can infer which task a knowledge worker is working on. This is a requirement for the context-aware functionality that we propose in this thesis, in order to support knowledge workers.

Limitations The limitation of the model we presented is that it is not validated with actual users. The main reason is that there is still not one accepted definition of what context actually entails. Furthermore, most attempts to capture context already involve assumptions about what belongs to the context and what not. And if simply everything would be measured, then the measuring activity itself becomes a part of the context as well. This makes it practically impossible to validate the model for context. The best we can do is to be thorough in the description of our assumptions, which is what we have done.

CHAPTER 6: AN INTERACTIVE ACTIVATION BASED MODEL FOR CONTEXT RECOGNITION AND IDENTIFICATION

Based on the conceptual and formal model in Chapter 5 we designed a novel algorithm for automatic context detection. In Chapter 6 we presented this algorithm. This algorithm, Contextual Interactive Activation (CIA), was based on the interactive activation model by McClelland and Rumelhart (1981). It consists of multiple layers: an input layer for observed events, a context layer to describe contextual information (topics, locations, time and entities) and a document layer to mimic the knowledge worker's existing knowledge. An activation function is used to propagate the network based on observed data. The observed data is presented in the form of event blocks;

all logged interactions of a user and his computer within a single computer window. Typically these event blocks cover a time period of only a few seconds.

The network can be used to categorize context into context categories by using a separate identification layer . As training material we used one example document for each category of interest. We showed that this approach is more effective in context identification and required less training effort then the traditional supervised methods k-NN and Naive Bayes. The limitation of the model is that the method for topic extraction used has a large influence on the overall performance of the model. On the positive side, the model does not limit the type of information that is represented, making it a flexible approach that could be used in a variety of applications. Furthermore there are many interesting aspects and opportunities of the model that have not been explored yet.

RQ 3. HOW CAN OUR CONCEPTUAL MODEL BE IMPLEMENTED AND HOW WELL CAN IT DETECT THE ACTIVE CONTEXT?

The algorithm that we designed and presented in Chapter 6 is capable of capturing context automatically with little user effort. In principle the model can be used in an unsupervised manner. The basis of the model is a network in which information is activated based on an observed event, and where activation is spread through the associations between information elements. The recognition of the context is in this case equal to the activation of the information elements in the context level of the network.

In the case of context identification, we can train the network to make associations between certain context labels and nodes in the context level of the network. For this purpose, we could make use of event blocks labelled with context labels to train the network. However, in order to reduce the user effort required, a method of transfer learning can be used. In this method, only one representative document per context label needs to be provided. The content of the documents is analysed to make the required associations between context level and context labels. Selecting a relevant document for a context label is much less effort than interpreting and labelling multiple event blocks. Although we could use this tactic for training traditional algorithms such as k-NN and Naive Bayes as well, these algorithms typically suffer from the difference between the source (documents) and target domain(event-blocks) (See Chapter 8).

Limitations A possible limitation of the implementation is that many aspects of the conceptual model, such as emotion or attention, are not actually implemented in the algorithm for context recognition and identification. There are two reasons. The first is that some aspects that we modelled in the conceptual model are hard to observe, such as which elements are consciously observed by the knowledge worker and which not. Therefore we simplified it by assuming that the knowledge worker observes everything that is visible.

The second reason is that some elements are not relevant for the applications that we describe in this thesis, or for which the relevance still needs to be validated. An example is the influence of emotions.

In terms of the effectiveness of the model in detecting context, the implementation is limited by our choice of entity extractor and topic extractor. There are many more options that can be explored. We are also limited by the evaluation data that we have available, as it does not contain long term data. Furthermore, we were not able to test the method on another dataset since there is no such dataset as far as we know. This makes it difficult to generalize the results.

PART 3: CONTEXT-AWARE SUPPORT

In part 3 of this thesis we described our research into context-aware support. The algorithms described in this part were centred around the goal to support knowledge workers during their work with automated categorization and recommendation software that require a minimal amount of effort from the knowledge worker to be used.

We first summarize the results of chapters 7 and 8 on experiments addressing e-mail categorizations with little user efforts. These categorizations can be seen as a method to add context to existing data, but they are also a prerequisite for context-aware notification filtering. We continue with an discussion on the question *"How can we reduce user effort in training algorithms for e-mail categorization?"*

Then, we summarize Chapter 9 about our experiments addressing context-aware document recommendation and its evaluation. This allows us to discuss the question *"How should we evaluate context-aware information recommendation and what are the benefits and downsides of various methods for context-aware information support?"*

We end with the summary of Chapter 10 on context-aware recommendation of touristic sights, which illustrates some interesting additional possibilities for context-aware recommendation systems.

In this part there are two main contributions. The first is a novel algorithm for e-mail categorization that reduces user labelling effort by making use of documents to categorize e-mail messages in user-defined categories; the folders of the documents. This algorithm is relevant for the field of Information Retrieval as a baseline for e-mail categorization methods with low user effort and high meaningfulness of the categories for the user.

The second contribution is a multi-faceted evaluation strategy of context-aware document recommendation for knowledge worker support. This is relevant for the field of Recommender Systems as their evaluation scenarios are typically less complex than the scenario of the knowledge worker.

CHAPTER 7: COMBINING TEXTUAL AND NON-TEXTUAL FEATURES FOR E-MAIL IMPORTANCE ESTIMATION

An important source of information overload is e-mail. One method to support a knowledge worker is to highlight those e-mail messages that are important for a knowledge worker. A reason that a message is considered to be important can be because it requires an activity from the user, such as a reply. In this chapter we have described an experiment in which we detect the reply expectation automatically. We used the messages that were replied to previously as training data, requiring no user effort. We compared three feature selection methods for reply prediction

with a Naive Bayes classifier or a Decision Tree. We concluded that the Naive Bayes classifier in combination with Linguistic Profiling as feature selection method had the best performance, with the additional advantage that it is a transparent method.

CHAPTER 8: E-MAIL CATEGORIZATION USING PARTIALLY RELATED TRAINING EXAMPLES

Another method for knowledge worker support is project-based e-mail categorization. This would allow the knowledge worker to filter all the messages that are not related to the project he is currently working on. In Chapter 8 we described an algorithm for e-mail categorization that was inspired by the context recognition algorithm from Chapter 6. It makes use of the same network and activation principles, but uses contextual information that is more focused on contact details, which is an important feature for e-mail categorization.

The algorithm also makes use of transfer learning by using foldered documents as training data. The assumption is that these folders with documents already exist, or can be made with less effort than labelling e-mail messages manually. The network algorithm achieved better accuracy than Naive Bayes,Linear SVM and k-NN baselines when trained on documents. Nevertheless, the accuracy was not optimal at 58%. The accuracy of the network could be increased to state-of-the-art level by additional training on labelled e-mail examples. The combination of using both documents and e-mail examples still reduces the effort for the user, as fewer examples are needed overall compared to the supervised approaches.

RQ 4. HOW CAN WE REDUCE USER EFFORT IN TRAINING ALGORITHMS FOR E-MAIL CATEGORIZATION?

In order to make use of machine learning techniques for e-mail message categorization there are multiple approaches that can be used to reduce user labelling effort while retaining the meaningfulness of categories for the user. A first approach to reduce user effort is to look at data that is already available. An example is the personal historic data on replied messages such as the data that was used in the reply prediction experiment in Chapter 6. This approach is especially suited for action prediction problems where actions are logged automatically. A limitation of this data is that the action-categories are not always the categories that are needed (are not useful or meaningful categories). For example, reply behaviour is only one factor in the categorization of interest; message priority.

Another possibility is to look at other sources. Sometimes there is another source with similar categories for which labelled data is already available, or for which the labels are easier to obtain. An example is the folder structure from Chapter 8, where we used the folder names as labels. When this is possible it is important to understand which features are necessary for the effective prediction of the category. If these features are different between the source and the target domain, algorithms need to be developed that improve the extraction of the correct features in the source domain. One such method is the network-based method described in Chapter 8.

A last possibility to reduce user effort is to use unsupervised methods for categorization. The problem with these methods, however, is that the meaningfulness and usefulness of the categories cannot be guaranteed.

Limitations The main limitation of the research is that we could not always quantify the reduction of user effort. Furthermore the concept of meaningfulness to the user should be investigated further: There are many categorizations that can be made and most of them are not actually used by the knowledge worker, even when they are meaningful. Therefore, we need to know which categories the knowledge worker would really benefit from.

Further limitations are that both categorization algorithms are evaluated on only one dataset. Therefore, we do not know how effective the algorithms are for other datasets. The reason that we have not evaluated on multiple datasets is that most datasets with e-mail messages are not shared in the community because of privacy concerns. And, when the dataset is open for the community, then the messages are not always annotated with the category of interest.

CHAPTER 9: EVALUATION OF CONTEXT-AWARE INFORMATION RECOMMENDATION SYSTEMS

Supporting a knowledge worker is a complex task and in fact requires a deep understanding of the user's activities, tasks and knowledge state. There are many aspects that play a role in determining whether the proposed support mechanism is effective. In Chapter 9 we described four evaluation criteria that capture the desired properties of context-aware information recommendation systems that were identified in a knowledge worker scenario: i) relevance of the information to the context, ii) relevance of the information to the task, iii) the possibility to predict which document will be opened next, and iv) the diversity of the recommendation lists.

Additionally we compared three methods for context-aware information recommendation: contextual pre-filtering in combination with content based recommendation (CBR), just-in-time information retrieval paradigm (JITIR) and our network-based approach where context is part of the recommendation model (CIA).

We concluded that each method has its own strengths. CBR is strong at context relevance, JITIR captures document relevance well and CIA achieves the best result at predicting user action. Overall, we concluded that the CIA approach is most suited for context-aware information recommendation for knowledge workers as it is the most flexible and robust in providing suggestions.

RQ 5. HOW SHOULD WE EVALUATE CONTEXT-AWARE INFORMATION RECOMMENDATION AND WHAT ARE THE BENEFITS AND DOWNSIDES OF VARIOUS METHODS FOR CONTEXT-AWARE INFORMATION SUPPORT?

Typical context-aware recommendation systems are evaluated mainly on relevance and predictive power (Ricci, Rokach, and Shapira, 2011). For the recommendation of information in the knowledge worker scenario, however, the requirements are a little different. First, there is a larger negative impact of irrelevant recommendations as these could distract the knowledge worker from his work. Secondly, the context of a knowledge worker is highly dynamic compared to for example a situation where a movie is recommended. Finally, there are stages in the knowledge worker's tasks where diversity of recommendations is important (exploration stage), or when it is important to simply re-find a single document that the user needs (finishing stage).

This means that in order to evaluate context-aware information recommendation in the knowledge worker scenario, multiple evaluation tasks need to be considered. Moreover it is important to understand the changing nature of the knowledge worker's intentions as they influence which evaluation task is most important.

Overall we have identified three families of context-aware information recommendation methods. The first is contextual pre-filtering in combination with content based recommendation (CBR). The advantage of this family is that it is really strong at relevance to the context, since contextual pre-filtering excludes information that does not have the right context. At the same time this is a weak point as it makes the approach inflexible. It requires a categorization of documents into contexts. Moreover it cannot use documents that are tagged with another context even though the topical content may overlap. Typically this type of algorithm depends on a manual source of active context selection. This means that a user needs to select the category of his current context first, before it can benefit from recommendations. This is not realistic in the dynamic knowledge worker context.

A second family is just-in-time information retrieval paradigm (JITIR). This method is capable of adjusting to the dynamic knowledge worker context as it uses the textual content that is visible on the screen as query for its suggestions. A downside of this approach is that in there is a strong focus on the active window. Although this means that recommendations will not likely be distracting, it also means that the recommendations by JITIR might be too focused to prove useful.

A final family is the recommendation approach where context is part of the recommendation model, such as our network model CIA. This method takes the dynamic nature of the context into account by using the active applications, typed text and so on as input. It also takes history into account, and uses a form of query expansion. This makes the model very flexible when there is little data available. Another advantage is that it provides diverse suggestions and is good at predicting which document is opened next. A downside is that this approach is more likely to recommend sources that do not strictly fit the context-category. This does not need to be a problem, provided that the recommended source is associated with a context-category that is topically related.

Limitations Again a limitation of this research is that there was only one dataset available that we could use. The development of new datasets is required in order to make more general claims concerning the relative performance of the systems.

Furthermore, there may be multiple metrics available for each evaluation task. We have not investigated alternative metrics for the tasks. In order to provide a proper guideline for the evaluation of context-aware document retrieval for knowledge worker a more thorough investigation of metrics should be done.

There are also some aspects in the conceptual context model that could be beneficial for context-aware document recommendation that were not taken into account in this research. An example is the use of facial expressions to understand whether a document gives a positive or negative emotion to the knowledge worker. This would give insight to the relevance of an observed document

Finally, for recommendation systems it is always desirable to perform an evalua-

tion with actual users, to see whether the documents that are relevant in theory are actually used in practice.

CHAPTER 10: RECOMMENDING PERSONALIZED TOURISTIC SIGHTS USING GOOGLE PLACES

In the final chapter of this thesis we have looked at context-aware recommendation of personalized touristic sights. This chapter shows the possibility to influence the user by combining personalized and non-personalized recommendations. Another method that was used to influence the user was to present the recommendation in a positive form based on positive research. This chapter illustrates that the design of a context-aware algorithm is not only concerned with finding appropriate recommendations, but could potentially also be used as persuasive technology. Using the algorithm could be a risk if used to support a knowledge worker, but at the same time gives the potential to protect the knowledge worker from his own bad behaviours. The algorithm could for example steer the knowledge worker away from documents that would distract him from his tasks

HOW CAN WE DESIGN, IMPLEMENT AND EVALUATE CONTEXT-AWARE METHODS THAT MAKE COMPUTER-BASED KNOWLEDGE WORK MORE EFFECTIVE AND MORE EFFICIENT?

In this thesis we have shown that there are many aspects that play a role in supporting knowledge workers with technology. Most importantly we should understand what the knowledge worker is doing and what his intentions are. Both of these elements are part of his context. A good understanding of this context is needed for the design of proper model for the knowledge worker's context.

The implementation of the context model into an algorithm for context recognition and identification should be flexible and dynamic. The network algorithm that we designed and implemented in this thesis, CIA, is suited for the task. It is flexible in the type of applications for which it can be used, such as context identification and context-aware recommendation and flexible in the type of contextual elements that can be used. Moreover it is highly dynamic as it uses computer interactions as input, and uses an activation function to spread through the network to activate all elements that can be associated with the input.

By using the CIA-model as a context-aware document recommendation system we can remind the knowledge worker of documents that can help him execute his tasks. This can help him make his work more effective as well as efficient, since information is available for re-use at any time.

Additionally the knowledge worker can be supposed to find and access messages more efficiently by categorizing his e-mail messages in a way that matches his intentions. One way is by categorizing messages by their project context, another by the activity they require. In the future, the combination of categorization of messages by context and priority could be used for context-aware notification filtering.

Limitations The main limitation of our research is that it is explorative in nature. It was difficult to find datasets to evaluate our work. This means that we typically had only one dataset for evaluation purposes, limiting the generalizability of our work.

In order to accelerate the research to knowledge worker support, more datasets need to be collected and prepared. For these datasets it is important that they reflect the complex nature of the knowledge worker's needs. Also, they should include a variety of sensors.

Another limitation is that we focused on data-driven research without looking at task lists etc. It is our goal to support knowledge workers, to reduce information overload and to improve well-being. In order to know whether our methods are effective, we should quantify and evaluate on information overload and well-being.

SUGGESTIONS FOR FUTURE WORK

Based on the limitations of the research in this thesis we offer some suggestions for future work. First, our work was mainly data-driven research. But, there were no datasets available containing the relevant data on the knowledge worker's life. Therefore, we had to collect our own datasets. In future work it is important that more datasets become available, in order to validate and generalize our findings.

Furthermore, the reduction of information overload and the improvement of mental well-being is our ultimate goal. Thus, to validate the effectiveness of our approaches, it is important to execute user studies. In these studies we can for example evaluate whether giving context-aware document recommendations actually reduces information overload.

Another aspect for future work is the contextual interactive activation model (CIA) itself. The model was now evaluated in a personal information management and recommendation setting. The design of the model, however, is suited for other fields as well. An example is the use of the approach to predict which aspects in the work of the knowledge worker are stressful. In that case connections between persons, topics and emotions can be made based on sensor data of facial expressions. This could help to model the emotional context of the knowledge worker and can be used to coach the knowledge worker on his week plan in order to spread stressful moments evenly.

In addition CIA could be extended with attentional aspects. An example could be by creating bias nodes, that either suppress or enhance activation of certain elements, based on the attention of the user. CIA could also be improved by optimizing the connection weights in the network using labelled data when it is available.

During the various experiments that were executed for this theses, we concluded that the context of a knowledge worker is diverse and dynamic. In order to support a knowledge worker, different support mechanisms are required for the different types of activities the knowledge worker engages in. The type of support that is needed can be dependent on the phase of the work. Moreover, not every knowledge worker appreciates the same kind of support suggesting the need for personalization. This means that it is unlikely that there is a single support algorithm that can do it all. In light of this intuition, it seems more realistic to imagine a suite of support algorithms within an application that supports the knowledge worker. Depending on the pref-

erences of the knowledge worker and the task that a knowledge worker is currently executing, the application selects the algorithm that provides the optimal support. In the case of document recommendation, for example, an algorithm that promotes variety in results could be selected when the knowledge worker is in the exploration phase of his research. When the knowledge worker is finishing his report, the application could select an algorithm that is more focused on relevance of the document.

In order to determine which algorithm is best suited at a certain moment in time, we propose to investigate a cost-benefit mechanism. This mechanism would weight the costs and the benefits for using each algorithm in the current situation, and would provide a ranking of which algorithm is best suited. In order to create such a selection mechanism, the costs in the knowledge worker's life should be quantified (e.g. cost of distraction). Furthermore, a quantification of the benefits of using a certain support algorithm should be investigated (e.g. benefit of a recommendation). These cost-benefit trade-offs can be personalized. Moreover, such a selection mechanism could be used in a persuasive style in order to nudge the knowledge worker into the right direction to improve his well-being.

NEDERLANDSE SAMENVATTING

Er is een toename in stress op het werk. Dit kan leiden tot gezondheidsklachten bij werknemers zoals een burn-out. In het SWELL project ontwikkelen en onderzoeken we ICT oplossingen die kenniswerkers kunnen ondersteunen in het bereiken van een gezonde leef- en werkstijl.

Eén van de oorzaken van stress op het werk is het probleem van "information overload". Door de beschikbaarheid van smartphones en tablets met continue internet-toegang worden individuen soms overspoeld met informatie. Het wordt moeilijker om werk en thuis te scheiden, maar het kan ook moeilijker worden om juist die informatie te vinden die je op een bepaald moment nodig hebt voor het goed uitoefenen van je werk.

Een mogelijke oplossing voor dit probleem is het creëren van applicaties die "context-aware" zijn. Dat wil zeggen dat deze applicaties een begrip hebben van wat een persoon aan het doen is, zodat ze op het juiste moment de juiste ondersteuning kunnen bieden in het werk. Dit is de achterliggende gedachte van mijn proefschrift geweest.

Dit proefschrift bestaat uit drie delen. In het eerste deel beantwoorden we de vraag "Hoe kunnen we context-aware methoden ontwerpen, ontwikkelen en evalueren die kantoorwerk effectiever en efficiënter maken?". Om deze vraag te beantwoorden hebben we kenniswerkers geobserveerd tijdens hun werk. We hebben hiermee data verzameld over hun werkgedrag. We kunnen op basis hiervan concluderen dat veel van de bedoelingen en doelen van een gebruiker impliciet aanwezig zijn, maar niet eenvoudig herkend kunnen worden. We kunnen wel een beeld krijgen van de gebruiker en zijn doelen door te kijken naar computer interacties zoals toetsaanslagen en muisklikken (de context informatie). Deze data biedt alleen wel een uitdaging, omdat het een grote hoeveelheid data is waarvan veel niet relevant.

In het tweede deel van mijn proefschrift modelleren we de context die nodig is om een kenniswerker te ondersteunen. Dit context model is dynamisch en wordt gedreven door gebeurtenissen in de omgeving van de kenniswerker. De kenniswerker bevindt zich in het centrum van de context, maar de context heeft ook een invloed op de kenniswerker zelf. We kunnen deze context observeren met behulp van sensoren die onafhankelijk zijn van de kenniswerker. Daar staat wel tegenover dat de interpretatie van de data uit deze sensoren alleen in samenhang met de kenniswerker begrepen kan worden.

Daarnaast beschrijven we in dit deel een algoritme, Contextual Interactive Activation (CIA), om automatisch de context van de kenniswerker te herkennen vanuit de interacties die een kenniswerker heeft met zijn pc. Dit algoritme kan puur vanuit de data, zonder tussenkomst van gebruikers, al iets zeggen over de context van de gebruiker. Daarnaast kan het met behulp van maar een paar voorbeelden van de gebruiker de context identificeren. Dat wil zeggen dat er een label gekoppeld wordt aan

de context, bijvoorbeeld "project Proefschrift".

Als laatste beschrijven we in deel drie van dit proefschrift algoritmen die kenniswerkers ondersteunen terwijl ze weinig input van de gebruiker vragen. Eén van de toepassingen waar we ons op gericht hebben is het categoriseren van e-mails. In de eerste applicatie delen we e-mail berichten in op basis van of ze beantwoord moeten worden of niet. In de tweede applicatie doen we de categorisatie op basis van het project waar de e-mail bij hoort. Voor deze e-mail categorisatie maken we gebruik van kennis over de gebruiker en zijn context. Hiervoor gebruiken we een aangepaste versie van het CIA-algoritme. Met CIA kunnen we even goed als bestaande methoden de berichten categoriseren maar hoeft de gebruiker minder voorbeelden te geven voordat de categorisatie gedaan kan worden.

In een laatste applicatie zetten we het CIA-algoritme in om informatie te zoeken, zonder dat de gebruiker hiervoor een zoekopdracht hoeft in te typen. We vergelijken de CIA-methode met andere, al bestaande, methoden om automatisch informatie aan te bevelen. Ook analyseren we de benodigdheden om een goede evaluatie te doen van dit soort aanbevelingssystemen voor kenniswerkers. Hierover kunnen we concluderen dat een multidimensionale evaluatie belangrijk is. Het blijkt dat elk van de methoden zijn eigen plus- en minpunten heeft en dat het afhankelijk is van de prioriteiten van de gebruiker welke methode het beste gebruikt kan worden.

De belangrijkste beperking van het onderzoek in dit proefschrift is dat het een exploratief onderwerp is. Er was weinig data beschikbaar om onze hypothesen en methoden te testen. Hierdoor kunnen we onze conclusies maar beperkt generaliseren. Dit laat ook zien dat het belangrijk is om meer datasets te verzamelen die de complexiteit van de kenniswerker, zijn taken, en zijn context reflecteren.

Toekomstig onderzoek moet uitwijzen wat het effect van het gebruik van context-aware applicaties is op de stress die de kenniswerker ervaart op zijn werk. Eén van de open vragen is welke ondersteunende applicatie op welk moment ingezet moet worden. Idealiter kunnen we met al deze technieken de kenniswerker zo helpen dat zijn mentale gezondheid vanzelf verbetert, zonder dat hij daarvoor actief moeite hoeft te doen.

BIBLIOGRAPHY

Abela, C., C. Staff, and S. Handschuh (2010). "Task-based user modelling for knowledge work support". In: *User Modeling, Adaptation, and Personalization*. Springer, pp. 419–422.

Aberdeen, D., O. Pacovsky, and A. Slater (2010). "The learning behind gmail priority inbox". In: *LCCC: NIPS 2010 Workshop on Learning on Cores, Clusters and Clouds*.

Akman, V. and M. Surav (1996). "Steps toward formalizing context". In: *AI magazine* 17.3, p. 55.

Allen, D. (2003). *Getting Things Done. The Art of Stress-Free Productivity*. ISBN-10: 0142000280 ISBN-13: 978-0142000281. Penguin. URL: `http://www.davidco.com/`.

Anderson, J. R. and G. H. Bower (1973). *Human associative memory*. VH Winston & Sons.

Ardissono, L., A. Goy, G. Petrone, M. Segnan, and P. Torasso (2003). "Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices". In: *Applied Artificial Intelligence* 17.8-9, pp. 687–714.

Ardissono, L. and G. Bosio (2012). "Context-dependent awareness support in open collaboration environments". In: *User Modeling and User-Adapted Interaction* 22.3, pp. 223–254.

Armentano, M. G. and A. A. Amandi (2012). "Modeling sequences of user actions for statistical goal recognition". In: *User Modeling and User-Adapted Interaction* 22.3, pp. 281–311.

Arnold, A., R. Nallapati, and W. W. Cohen (2007). "A comparative study of methods for transductive transfer learning". In: *Data Mining (ICDM)*. IEEE, pp. 77–82.

Ayodele, T. and S. Zhou (2009). "Applying machine learning techniques for e-mail management: solution with intelligent e-mail reply prediction". In: *Journal of Engineering and Technology Research* 1.7, pp. 143–151.

Baeza-Yates, R., L. Calderón-Benavides, and C. González-Caro (2006). "The Intention Behind Web Queries". In: *String Processing and Information Retrieval*. Ed. by F. Crestani, P. Ferragina, and M. Sanderson. LNCS 4209. Berlin Heidelberg: Springer-Verlag, pp. 98–109.

Bahadori, M. T., Y. Liu, and D. Zhang (2011). "Learning with minimum supervision: A general framework for transductive transfer learning". In: *Data Mining (ICDM)*. IEEE, pp. 61–70.

Bauer, T. and D. Leake (2001). "Word Sieve: A Method for Real-Time Context Extraction". In: *Modeling and Using Context*, pp. 30–44.

Bawden, D. and L. Robinson (2009). "The dark side of information: overload, anxiety and other paradoxes and pathologies". In: *Journal of information science* 35.2, pp. 180–191.

Bekkerman, R. (2004). "Automatic categorization of email into folders: Benchmark experiments on Enron and SRI corpora". In: *Computer Science Department Faculty Publication Series*, p. 218.

Benselin, J. C. and G. Ragsdell (2015). "Information overload: The differences that age makes". In: *Journal of Librarianship and Information Science*, p. 0961000614566341.

Biedert, R., S. Schwarz, and T. Roth-Berghofer (2008). "Designing a Context-sensitive Dashboard for Adaptive Knowledge Worker Assistance". In:

Blanc-Brude, T. and D. L. Scapin (2007). "What do people recall about their documents?: implications for desktop search tools". In: *Proceedings of the 12th international conference on Intelligent user interfaces*. ACM, pp. 102–111.

Blatter, B., I. Houtman, S. van den Bossche, K. Kraan, and S. van den Heuvel (2005). *Gezondheidsschade en kosten als gevolg van RSI en psychosociale arbeidsbelasting in Nederland*. TNO Kwaliteit van leven.

Brdiczka, O. (2010). "From documents to tasks: deriving user tasks from document usage patterns". In: *Proceedings of the 15th international conference on Intelligent user interfaces*. ACM, pp. 285–288.

Broder, A. (2002). "A taxonomy of web search". In: *ACM Sigir forum*. Vol. 36. ACM, pp. 3–10.

Budzik, J. and K. J. Hammond (2000). "User interactions with everyday applications as context for just-in-time information access". In: *Proceedings of the 5th international conference on intelligent user interfaces*. ACM, pp. 44–51.

Buriano, L. (2006). "Exploiting social context information in context-aware mobile tourism guides". In: *Proc. of Mobile Guide 2006*.

Cai, F., S. Liang, and M. de Rijke (2014). "Personalized Document Re-ranking Based on Bayesian Probabilistic Matrix Factorization". In: *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*. SIGIR '14. Gold Coast, Queensland, Australia: ACM, pp. 835–838.

Calderón-Benavides, L., C. González-Caro, and R. Baeza-Yates (2010). "Towards a Deeper Understanding of the User's Query Intent". In: *Workshop on Query Representation and Understanding, SIGIR 2010*, pp. 21–24.

Carpineto, C., R. de Mori, G. Romano, and B. Bigi (2001). "An information-theoretic approach to automatic query expansion". In: *ACM Trans. Inf. Syst.* 19.1, pp. 1–27. ISSN: 1046-8188. DOI: 10.1145/366836.366860. URL: http://doi.acm.org/10.1145/366836.366860.

Carvalho, V. R. and W. W. Cohen (2005). "On the collective classification of email speech acts". In: *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 345–352.

Cecchinato, M., A. L. Cox, and J. Bird (2014). "I Check My Email on the Toilet': Email Proactices and Work-Home Boundary Management". In: *Proceedngs of the MobileHCI Workshop*.

Chakravarthy, S., A. Venkatachalam, and A. Telang (2010). "A graph-based approach for multi-folder email classification". In: *Data Mining (ICDM), 2010 IEEE 10th International Conference on*. IEEE, pp. 78–87.

Chen, Y. and G. J. F. Jones (2014). "Are Episodic Context Features Helpful for Refinding Tasks?: Lessons Learnt from a Case Study with Lifelogs". In: *Proceedings of the 5th Information Interaction in Context Symposium*. IIiX '14. ACM, pp. 76–85.

Cheyer, A., J. Park, and R. Giuli (2005). "Iris: Integrate. relate. infer. share". In: *Semantic Desktop Workshop*. Citeseer.

Cohen, J (1968). "Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit." In: *Psychological Bulletin* 70.4, pp. 213–220.

Cohen, W. W., V. R. Carvalho, and T. M. Mitchell (2004). "Learning to Classify Email into"Speech Acts"." In: *EMNLP*, pp. 309–316.

Dabbish, L., R. Kraut, S. Fussell, and S. Kiesler (2004). "To reply or not to reply: Predicting action on an email message". In: *ACM 2004 Conference*. Citeseer.

Dabbish, L., R. Kraut, S. Fussell, and S. Kiesler (2005). "Understanding email use: predicting action on a message". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, pp. 691–700.

Dankbaar, B. and G. Vissers (2009). "Of knowledge and work". In:

Davenport, T. H. (2013). *Thinking for a living: how to get better performances and results from knowledge workers*. Harvard Business Press.

Deepak, P., D. Garg, and V. Varshney (2007). "Analysis of Enron Email Threads and Quantification of Employee Responsiveness". In: *Workshop on Text Mining and Link Analysis (TextLink 2007)*.

Demerouti, E., A. B. Bakker, F. Nachreiner, and W. B. Schaufeli (2001). "The job demands-resources model of burnout." In: *Journal of Applied psychology* 86.3, p. 499.

Dervin, B. (1997). "Given a context by any other name: Methodological tools for taming the unruly beast". In: *Information seeking in context* 13, p. 38.

Devaurs, D., A. S. Rath, and S. N. Lindstaedt (2012). "Exploiting the user interaction context for automatic task detection". In: *Applied Artificial Intelligence* 26.1-2, pp. 58–80.

Dey, A. K., G. D. Abowd, and D. Salber (2001). "A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications". In: *Human-computer interaction* 16.2, pp. 97–166.

Dietterich, T. G. (1998). "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms". In: *Neural Computation* 10, pp. 1895–1923.

Dijkstra, T., W. J. Van Heuven, and J. Grainger (1998). "Simulating cross-language competition with the bilingual interactive activation model." In: *Psychologica Belgica*.

Dourish, P. (2004). "What we talk about when we talk about context". In: *Personal and ubiquitous computing* 8.1, pp. 19–30.

Dredze, M., J. Blitzer, and F. Pereira (2005). "Reply expectation prediction for email management". In: *The Second Conference on Email and Anti-Spam (CEAS), Stanford, CA*.

Dredze, M., T. Brooks, J. Carroll, J. Magarick, J. Blitzer, and F. Pereira (2008). "Intelligent email: Reply and attachment prediction". In: *Proceedings of the 13th international conference on Intelligent user interfaces*. ACM, pp. 321–324.

Dredze, M., T. Lau, and N. Kushmerick (2006). "Automatically classifying emails into activities". In: *Proceedings of the 11th international conference on Intelligent user interfaces*. ACM, pp. 70–77.

Dumais, S., E. Cutrell, J. Cadiz, G. Jancke, R. Sarin, and D. Robbins (2003). "Stuff I've seen: a system for personal information retrieval and re-use". In: *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. ACM, pp. 72–79.

Dumais, S., E. Cutrell, R. Sarin, and E. Horvitz (2004). "Implicit queries (IQ) for contextualized search". In: *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 594–594.

Elsweiler, D., I. Ruthven, and C. Jones (2007). "Towards memory supporting personal information management tools". In: *Journal of the American Society for Information Science and Technology* 58.7, pp. 924–946.

Ermolayev, V., C Ruiz, M. Tilly, E. Jentzsch, J. M. Gomez-Perez, and W.-E. Matzke (2010). "A context model for knowledge workers". In: *Proceedings of the 2nd Workshop on Context, Information and Ontologies*. Vol. 626. Impact Information.

Finkel, J. R., T. Grenager, and C. Manning (2005). "Incorporating non-local information into information extraction systems by gibbs sampling". In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, pp. 363–370.

Gains, J. (1999). "Electronic mail—a new style of communication or just a new medium?: An investigation into the text features of e-mail". In: *English for specific purposes* 18.1, pp. 81–101.

Gantz, J., A. Boyd, and S. Dowling (2009). "Cutting the clutter: Tackling information overload at the source". In: *International Data Corporation White Paper*.

Gao, A. and D. Bridge (2010). "Using shallow natural language processing in a just-in-time information retrieval assistant for bloggers". In: *Artificial Intelligence and Cognitive Science*. Springer, pp. 103–113.

Gayo-Avello, D. (2009). "A survey on session detection methods in query logs and a proposal for future evaluation". In: *Information Sciences* 179, pp. 1822–1843.

Ghadessy, M. and J. Webster (1988). "Form and function in English business letters: implications for computer-based learning". In: *Registers of written English: situational factors and linguistic features*, pp. 110–127.

Gomez-Perez, J., M. Grobelnik, C. Ruiz, M. Tilly, and P. Warren (2009). "Using task context to achieve effective information delivery". In: *Proceedings of the 1st Workshop on Context, Information and Ontologies*, p. 3.

González-Caro, C., L. Calderón-Benavides, R. Baeza-Yates, L. Tansini, and D. Dubhashi (2011). "Web Queries: the Tip of the Iceberg of the User's Intent". In: *Workshop on User Modeling for Web Applications, WSDM 2011*.

Granitzer, M., A. S Rath, M. Kröll, C. Seifert, D. Ipsmiller, D. Devaurs, N. Weber, and S. Lindstaedt (2009). "Machine learning based work task classification". In: *Journal of Digital Information Management* 7.5, pp. 306–314.

Grbovic, M., G. Halawi, Z. Karnin, and Y. Maarek (2014). "How Many Folders Do You Really Need?: Classifying Email into a Handful of Categories". In: *Proceedings of*

the 23rd ACM International Conference on Conference on Information and Knowledge Management. ACM, pp. 869–878.

Grossberg, S. (1976). "Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors". In: *Biological cybernetics* 23.3, pp. 121–134.

Groza, T., S. Handschuh, K. Moeller, G. Grimnes, L. Sauermann, E. Minack, M. Jazayeri, C. Mesnage, G. Reif, and R. Gudjonsdottir (2007). "The NEPOMUK Project-On the way to the Social Semantic Desktop". In: *Proceedings of the 3rd International Conference on Semantic Technologies*. Graz University of Technology, pp. 201–210.

Guo, Q. and E. Agichtein (2012). "Beyond dwell time: estimating document relevance from cursor movements and other post-click searcher behavior". In: *Proceedings of the 21st international conference on World Wide Web*. ACM, pp. 569–578.

Gurrin, C., A. F. Smeaton, and A. R. Doherty (2014). "LifeLogging: personal big data". In: *Foundations and Trends in Information Retrieval* 8.1, pp. 1–125.

Gyllstrom, K. and C. Soules (2008). "Seeing is retrieving: building information context from what the user sees". In: *Proceedings of the 13th international conference on Intelligent user interfaces*. ACM, pp. 189–198.

Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten (2009). "The WEKA data mining software: an update". In: *ACM SIGKDD Explorations Newsletter* 11.1, pp. 10–18.

Hanrahan, B. V., M. A. Pérez-Quiñones, and D. Martin (2014). "Attending to Email". In: *Interacting with Computers*, iwu048.

Henzinger, M., B.-W. Chang, B. Milch, and S. Brin (2005). "Query-free news search". In: *World Wide Web* 8.2, pp. 101–126.

Hinne, M., M. van der Heijden, S. Verberne, and W. Kraaij (2011). "A multi-dimensional model for search intent". In: *Proceedings of the Dutch-Belgium Information Retrieval workshop (DIR 2011)*, pp. 20–24.

Ho, J. and R. Tang (2001). "Towards an optimal resolution to information overload: an infomediary approach". In: *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*. ACM, pp. 91–96.

Huang, Y. and T. M. Mitchell (2008). "Exploring hierarchical user feedback in email clustering". In: *Email'08: Proceedings of the Workshop on Enhanced Messaging-AAAI*, pp. 36–41.

Ingwersen, P. and K. Järvelin (2005). *The turn: Integration of information seeking and retrieval in context*. Vol. 18. Springer.

Kalia, A., H. R. Motahari Nezhad, C. Bartolini, and M. Singh (2013). *Identifying business tasks and commitments from email and chat conversations*. Tech. rep. tech. report, HP Labs.

Kalman, Y. M. and G. Ravid (2015). "Filing, piling, and everything in between: The dynamics of E-mail inbox management". In: *Journal of the Association for Information Science and Technology*.

Karatzoglou, A., X. Amatriain, L. Baltrunas, and N. Oliver (2010). "Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering". In: *Proceedings of the fourth ACM conference on Recommender systems*. ACM, pp. 79–86.

Kelly, D. (2009). "Methods for evaluating interactive information retrieval systems with users". In: *Foundations and Trends in Information Retrieval* 3.1—2, pp. 1–224.

Kelly, L., Y. Chen, M. Fuller, and G. J. Jones (2008). "A study of remembered context for information access from personal digital archives". In: *Proceedings of the second international symposium on Information interaction in context*. ACM, pp. 44–50.

Kendall, M. G. (1938). "A new measure of rank correlation". In: *Biometrika* 30.1/2, pp. 81–93.

Kersten, M. and G. C. Murphy (2012). "Task Context for Knowledge Workers". In: *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*.

Kim, Y., A. Hassan, R. W. White, and I. Zitouni (2014). "Modeling dwell time to predict click-level satisfaction". In: *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, pp. 193–202.

Kiritchenko, S. and S. Matwin (2001). "Email classification with co-training". In: *Proceedings of the 2001 conference of the Centre for Advanced Studies on Collaborative research*. Citeseer, p. 8.

Klimt, B. and Y. Yang (2004). "The enron corpus: A new dataset for email classification research". In: *Machine Learning: ECML 2004*, pp. 217–226.

Koldijk, S, M Neerincx, and W Kraaij (2012). "Unobtrusively measuring stress and workload of knowledge workers". In: *Proceedings of Measuring Behavior*.

Koldijk, S., M. van Staalduinen, M. Neerincx, and W. Kraaij (2012). "Real-time task recognition based on knowledge workers' computer activities". In: *Proceedings of ECCE 2012 (Edinburgh, Scotland)*.

Koldijk, S., M. Sappelli, M. Neerincx, and W. Kraaij (2013). "Unobtrusive monitoring of knowledge workers for stress self-regulation". In: *Proceedings of the 21th International Conference on User Modeling, Adaptation and Personalization*.

Koldijk, S., M. Sappelli, S. Verberne, M. A. Neerincx, and W. Kraaij (2014). "The SWELL Knowledge Work Dataset for Stress and User Modeling Research". In: *Proceedings of the 16th International Conference on Multimodal Interaction*. ACM, pp. 291–298.

Kooti, F., L. M. Aiello, M. Grbovic, K. Lerman, and A. Mantrach (2015). "Evolution of Conversations in the Age of Email Overload". In: *Proceedings of the 24th International World Wide Web Conference*.

Koren, Y., E. Liberty, Y. Maarek, and R. Sandler (2011). "Automatically tagging email by leveraging other users' folders". In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 913–921.

Krzywicki, A. and W. Wobcke (2010). "Exploiting concept clumping for efficient incremental e-mail categorization". In: *Advanced Data Mining and Applications*. Springer, pp. 244–258.

Kulkarni, A. and T. Pedersen (2005). "SenseClusters: Unsupervised Clustering and Labeling of Similar Contexts". In: *Proceedings of the ACL 2005 on Interactive Poster and Demonstration Sessions*. ACLdemo '05. Ann Arbor, Michigan: Association for Computational Linguistics, pp. 105–108. DOI: 10.3115/1225753.1225780. URL: http://dx.doi.org/10.3115/1225753.1225780.

Kullback, S. and R. Leibler (1951). "On information and sufficiency". In: *The Annals of Mathematical Statistics* 22.1, pp. 79–86.

Lai, C.-H., D.-R. Liu, and C.-S. Lin (2013). "Novel personal and group-based trust models in collaborative filtering for document recommendation". In: *Information Sciences* 239, pp. 31–49.

Lakiotaki, K., N. F. Matsatsinis, and A. Tsoukias (2011). "Multicriteria User Modeling in Recommender Systems". In: *IEEE Intelligent Systems* 26.2, pp. 64–76.

Lampert, A., R. Dale, and C. Paris (2008). "Requests and Commitments in Email are More Complex Than You Think: Eight Reasons to be Cautious". In: *Australasian Language Technology Association Workshop 2008*. Vol. 6, pp. 64–72.

Landis, J. R. and G. G. Koch (1977). "The measurement of observer agreement for categorical data". In: *biometrics*, pp. 159–174.

Lansdale, M. (1988). "The psychology of personal information management". In: *Applied Ergonomics* 19.1, pp. 55–66.

Lehmann, J., M. Lalmas, G. Dupret, and R. Baeza-Yates (2013). "Online multitasking and user engagement". In: *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. ACM, pp. 519–528.

Lin, C.-Y. (2004). "Rouge: A package for automatic evaluation of summaries". In: *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pp. 74–81.

Maslach, C. and M. P. Leiter (2008). "Early predictors of job burnout and engagement." In: *Journal of applied psychology* 93.3, p. 498.

Maus, H. (2001). "Workflow context as a means for intelligent information support". In: *Modeling and Using Context*, pp. 261–274.

Maus, H., S. Schwarz, J. Haas, and A. Dengel (2011). "CONTASK: Context-Sensitive Task Assistance in the Semantic Desktop". In: *Enterprise Information Systems*. Springer, pp. 177–192.

McCallum, A. K. (2002). "MALLET: A Machine Learning for Language Toolkit". http://mallet.cs.umass.edu.

McClelland, J. L. and D. E. Rumelhart (1981). "An interactive activation model of context effects in letter perception: I. An account of basic findings." In: *Psychological review* 88.5, p. 375.

Melguizo, M. C. P., T. Bajo, and O. Gracia Castillo (2010). "A Proactive Recommendation System for Writing in the Internet Age". In: *Journal of Writing Research* 2.1.

Misra, S. and D. Stokols (2011). "Psychological and health outcomes of perceived information overload". In: *Environment and behavior*, p. 0013916511404408.

Oku, K., S. Nakajima, J. Miyazaki, and S. Uemura (2006). "Context-Aware SVM for Context-Dependent Information Recommendation". In: *Proceedings of the 7th International Conference on Mobile Data Management*. MDM '06. Washington, DC, USA: IEEE Computer Society, pp. 109–.

Oliver, N., G. Smith, C. Thakkar, and A. C Surendran (2006). "SWISH: semantic analysis of window titles and switching history". In: *Proceedings of the 11th IUI conference*. ACM, pp. 194–201. ISBN: 1595932879.

Omata, M., K. Ogasawara, and A. Imamiya (2010). "A project restarting support system using the historical log of a userś window usage". In: *Proceedings of the 22nd SIGCHI conference*. ACM, pp. 25–32.

On, B., E. Lim, J. Jiang, A. Purandare, and L. Teow (2010). "Mining interaction behaviors for email reply order prediction". In: *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*. IEEE, pp. 306–310.

Pan, S. J. and Q. Yang (2010). "A survey on transfer learning". In: *Knowledge and Data Engineering* 22.10, pp. 1345–1359.

Park, S. and D. U. An (2010). "Automatic E-mail Classification Using Dynamic Category Hierarchy and Semantic Features." In: *IETE Technical Review* 27.6.

Penco, C. (1999). "Objective and cognitive context". In: *Modeling and Using Context*. Springer, pp. 270–283.

Peterson, K., M. Hohensee, and F. Xia (2011). "Email formality in the workplace: A case study on the Enron corpus". In: *Proceedings of the Workshop on Languages in Social Media*. Association for Computational Linguistics, pp. 86–95.

Radicati, S. (2010). *Email Statistics Report*. URL: `http://www.radicati.com/wp/wp-content/uploads/2010/04/Email-Statistics-Report-2010-2014-Executive-Summary2.pdf` (visited on 04/11/2014).

Rath, A. S., D. Devaurs, and S. N. Lindstaedt (2010). "Studying the factors influencing automatic user task detection on the computer desktop". In: *Sustaining TEL: From Innovation to Learning and Practice*. Springer, pp. 292–307.

Rendle, S., Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme (2011). "Fast context-aware recommendations with factorization machines". In: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, pp. 635–644.

Reuters (1998). *Out of the Abyss: Surviving the information age*.

Rhodes, B. J. (1997). "The wearable remembrance agent: A system for augmented memory". In: *Personal Technologies* 1.4, pp. 218–224.

Ricci, F., L. Rokach, and B. Shapira (2011). *Introduction to recommender systems handbook*. Springer.

Rose, D. and D. Levinson (2004). "Understanding user goals in web search". In: *Proceedings of the 13th international conference on World Wide Web (WWW 2004)*. ACM, pp. 13–19.

Ruff, J. (2002). "Information Overload: Causes, Symptoms and Solutions". In: *Harvard Graduate School of Education*, pp. 1–13.

Sahami, M., S. Dumais, D. Heckerman, and E. Horvitz (1998). "A Bayesian approach to filtering junk e-mail". In: *Learning for Text Categorization: Papers from the 1998 workshop*. Vol. 62, pp. 98–105.

Salton, G. and C. Buckley (1988). "Term-weighting approaches in automatic text retrieval". In: *Information processing & management* 24.5, pp. 513–523.

Sappelli, M., S. Verberne, and W. Kraaij (2012). "Using file system content to organize e-mail". In: *Proceedings of the fourth symposium on Information interaction in context*.

Sappelli, M., S. Verberne, and W. Kraaij (2013a). "Combining textual and non-textual features for e-mail importance estimation". In: *Proceedings of the 25th Benelux Conference on Artificial Intelligence*.

Sappelli, M., S. Verberne, and W. Kraaij (2013b). "TNO and RUN at the TREC 2012 Contextual Suggestion Track: Recommending personalized touristic sights using

Google Places". In: *21st Text REtrieval Conference Notebook Proceedings (TREC 2012)*.

Sappelli, M., S. Verberne, and W. Kraaij (2014). "E-mail categorization using partially related training examples". In: *Proceedings of the 5th Information Interaction in Context Symposium*.

Sappelli, M., S. Verberne, K. S.J., and W. Kraaij (2014). "Collecting a dataset of information behaviour in context". In: *Proceedings of the 4th Workshop on Context-awareness in Retrieval and Recommendation*.

Sappelli, M., S. Verberne, and W. Kraaij (2013c). "Recommending personalized touristic sights using google places". In: *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. SIGIR '13. Dublin, Ireland: ACM, pp. 781–784. ISBN: 978-1-4503-2034-4. DOI: 10.1145/2484028.2484155. URL: http://doi.acm.org/10.1145/2484028.2484155.

Schick, A., L. Gordon, and S. Haka (1990). "Information overload: A temporal approach". In: *Accounting, Organizations and Society* 15.3, pp. 199–220.

Schwinger, W., C. Grün, B. Pröll, W. Retschitzegger, and A. Schauerhuber (2005). "Context-awareness in mobile tourism guides–A comprehensive survey". In: *Rapport Technique. Johannes Kepler University Linz*.

Segal, R. B. and J. O. Kephart (1999). "MailCat: An Intelligent Assistant for Organizing e-Mail". In: *Proceedings of the Third Annual Conference on Autonomous Agents*. AGENTS '99. Seattle, Washington, USA: ACM, pp. 276–282. ISBN: 1-58113-066-X. DOI: 10.1145/301136.301209. URL: http://doi.acm.org/10.1145/301136.301209.

Shen, J., L. Li, T. G Dietterich, and J. L Herlocker (2006). "A hybrid learning system for recognizing user tasks from desktop activities and email messages". In: *Proceedings of the 11th international conference on Intelligent user interfaces*. ACM, pp. 86–92. ISBN: 1595932879.

Silvestri, F. (2010). "Mining query logs: Turning search usage data into knowledge". In: *Foundations and Trends in Information Retrieval* 4.1-2, pp. 1–174. ISSN: 1554-0669.

Spira, J. and D. Goldes (2007). "Information overload: We have met the enemy and he is us". In: *Basex Inc*.

Štajner, T., D. Mladenić, and M. Grobelnik (2010). "Exploring contexts and actions in knowledge processes". In: *Proceedings of the 2nd International Workshop on Context, Information and Ontologies*.

Stumpf, S., X. Bao, A. Dragunov, T. G. Dietterich, J. Herlocker, K. Johnsrude, L. Li, and J Shen (2005). "Predicting user tasks: I know what youŕe doing". In: *20th National Conference on Artificial Intelligence, Workshop on Human Comprehensible Machine Learning*.

Sushmita, S., B. Piwowarski, and M. Lalmas (2010). "Dynamics of genre and domain intents". In: *Information Retrieval Technology*, pp. 399–409.

Tyler, J. and J. Tang (2003). "When can I expect an email response? A study of rhythms in email usage". In: *Proceedings of the eighth conference on European Conference*

on Computer Supported Cooperative Work. Kluwer Academic Publishers, pp. 239–258.

Van Halteren, H. (2004). "Linguistic profiling for author recognition and verification". In: *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, p. 199.

Venolia, G. and C. Neustaedter (2003). "Understanding sequence and reply relationships within email conversations: a mixed-model visualization". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, pp. 361–368.

Verberne, S. and M. Sappelli (2013). *SWELL D3.2 Activity classification*. Tech. rep. Available on `http://www.swell-project.net/results/deliverables`. COMMIT P7 SWELL.

Verberne, S., M. Sappelli, and W. Kraaij (2013). "Term extraction for user profiling: evaluation by the user". In: *Proceedings of the 21th International Conference on User Modeling, Adaptation and Personalization*.

Verberne, S., M. Heijden, M. Hinne, M. Sappelli, S. Koldijk, E. Hoenkamp, and W. Kraaij (2013). "Reliability and validity of query intent assessments". In: *Journal of the American Society for Information Science and Technology* 64.11, pp. 2224–2237.

Wakeling, S., P. Clough, and B. Sen (2014). "Investigating the Potential Impact of Non-personalized Recommendations in the OPAC: Amazon vs. WorldCat.Org". In: *Proceedings of the 5th Information Interaction in Context Symposium*. ACM, pp. 96–105.

Warren, P., J. M Gomez-Perez, C. Ruiz, J. Davies, I. Thurlow, and I. Dolinsek (2010). "Context as a tool for organizing and sharing knowledge". In: *Proceedings of the 2nd International Workshop on Context, Information and Ontologies*.

Warren, P. (2013). "Personal Information Management: The Case for an Evolutionary Approach". In: *Interacting with Computers*.

Webber, W., A. Moffat, and J. Zobel (2010). "A similarity measure for indefinite rankings". In: *ACM Transactions on Information Systems (TOIS)* 28.4, p. 20.

Weng, S.-S. and H.-L. Chang (2008). "Using ontology network analysis for research document recommendation". In: *Expert Systems with Applications* 34.3, pp. 1857–1869.

White, R., P. Bennett, and S. Dumais (2010). "Predicting short-term interests using activity-based search context". In: *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, pp. 1009–1018.

Whiting, S. and J. Jose (2011). "Context modelling for situation-sensitive recommendations". In: *Flexible Query Answering Systems*. Springer, pp. 376–387.

Whittaker, S. and C. Sidner (1996). "Email overload: exploring personal information management of email". In: *Proceedings of the SIGCHI conference on Human factors in computing systems: common ground*. ACM, pp. 276–283.

Xiang, Y. (2009). "Managing email overload with an automatic nonparametric clustering system". In: *The Journal of Supercomputing* 48.3, pp. 227–242.

Yang, Y. and J. Pedersen (1997). "A comparative study on feature selection in text categorization". In: *ICML*. Morgan Kaufmann Publishers, Inc., pp. 412–420.

CURRICULUM VITÆ

Maya Sappelli was born in Eindhoven in 1988. She followed her primary education in Son. In 2000 she moved to Well, where she followed her secondary education at the Raayland College in Venray. She graduated with a curriculum that was centred around language and culture, which she extended with courses on Biology and Computer Science.

From 2005 to 2009 Maya studied Artificial Intelligence. From 2006 to 2009 she pursued a bachelor in Linguistics as well. She graduated both curricula in 2009 with a combined thesis titled "Similarity-dependent cognate inhibition effects". In 2009, she continued her education in the master program of Artificial Intelligence with a specialization in Cognitive Research and Cognitive Engineering. She took a special interest in Language and Speech Technology. Her master project was executed at Philips Research and resulted in the thesis "Meal recommendation for Diabetes Type II patients".

Immediately after her graduation in 2011 she continued as a junior researcher at TNO in the COMMIT-project Smart Reasoning for well-being at work and at home (SWELL). She worked on the topic of supporting knowledge workers during their work with context-aware applications that reduce information overload, which has resulted in this thesis. Maya continues to work as a media-mining researcher at TNO.

LIST OF PUBLICATIONS

1. **Maya Sappelli**, Suzan Verberne, Gabriella Pasi, Maaike de Boer, Wessel Kraaij (2016) *Collecting tasks and intent of e-mail messages*, Under revision: Information Sciences.

2. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2016) *Evaluation of context-aware recommendation systems for information re-finding*, Accepted for publication in: Journal of the American Society for Information Science and Technology.

3. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2016) *Adapting the interactive activation model for context recognition and identification*, Under revision: ACM Transactions on Interactive Intelligent Systems.

4. Klamer Schutte, Henri Bouma, John Schavemaker, Laura Daniele, **Maya Sappelli**, Gijs Koot, Pieter Eendebak, George Azzopardi, Martijn Spitters, Maaike de Boer, Maarten Kruithof, Paul Brandt (2015) *Interactive detection of incrementally learned concepts in images with ranking and semantic query interpretation*, Proceedings of Content-Based Multimedia Indexing Workshop (CBMI 2015).

5. Maaike de Boer, Laura Daniele, Paul Brandt, **Maya Sappelli** (2015) *Applying Semantic Reasoning in Image Retrieval*, Proceedings of the International Workshop on Knowledge Extraction and Semantic Annotation (KESA 2015) – Best Paper.

6. Suzan Verberne, **Maya Sappelli**, Kalervo Järvelin, Wessel Kraaij (2015) *User simulations for interactive search: evaluating personalized query suggestion*, Proceedings of the 37th European Conference on Information Retrieval (ECIR 2015).

7. Saskia Koldijk, **Maya Sappelli**,Suzan Verberne, Mark Neerincx, Wessel Kraaij (2014) *The SWELL Knowledge Work Dataset for Stress and User Modeling Research*, Proceedings of the International Conference on Multimodal Interaction.

8. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2014) *E-mail categorization using partially related training examples*, Proceedings of the 5th Information Interaction in Context Symposium (IIiX 2014).

9. John Schavemaker, Erik Boertjes, Saskia Koldijk, Leon Wiertz, Suzan Verberne, **Maya Sappelli**, Rianne Kaptein (2014) *Fishualization: a group feedback display*, Proceedings of Measuring Behavior 2014.

10. **Maya Sappelli**, Suzan Verberne, Saskia Koldijk, Wessel Kraaij (2014) *Collecting a dataset of information behaviour in context.*, Proceedings of the 4th Workshop on Context-awareness in Retrieval and Recommendation (CARR @ ECIR 2014).

11. Suzan Verberne, **Maya Sappelli**, Wessel Kraaij (2014) *Query term suggestion in academic search*, Proceedings of the 36th European Conference on Information Retrieval (ECIR 2014).

12. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2013) *Combining textual and non-textual features for e-mail importance estimation*, Proceedings of the 25th Benelux Conference on Artificial Intelligence (BNAIC 2013).

13. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2013) *Recommending personalized touristic Sights using Google Places*, Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2013), Dublin (poster).

14. **Maya Sappelli** (2013) *The role of current working context in Professional Search*, Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2013), Dublin (doctoral consortium).

15. Suzan Verberne, **Maya Sappelli**, Wessel Kraaij (2013) *Term extraction for user profiling: evaluation by the user.*, Proceedings of the 21th International Conference on User Modeling, Adaptation and Personalization (UMAP 2013).

16. Saskia Koldijk, **Maya Sappelli**, Mark Neerincx, Wessel Kraaij (2013) *Unobtrusive monitoring of knowledge workers for stress self-regulation*, Proceedings of the 21th International Conference on User Modeling, Adaptation and Personalization (UMAP 2013).

17. Suzan Verberne, **Maya Sappelli**, Diana Ransgaard Sørensen, Wessel Kraaij (2013) *Personalization in Professional Academic Search.*, Proceedings of the Workshop on Integrating IR technologies for Professional Search (IRPS 2013).

18. Suzan Verberne, Maarten van der Heijden, Max Hinne, **Maya Sappelli**, Eduard Hoenkamp, Saskia Koldijk, Wessel Kraaij (2013) *Reliability and Validity of Query Intent Assessments*, Journal of the American Society for Information Science and Technology – Best Paper.

19. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2012) *TNO and RUN at the TREC 2012 Contextual Suggestion Track: Recommending personalized touristic sights using Google Places*, 21st Text REtrieval Conference Notebook Proceedings (TREC 2012).

20. Wouter Bokhove, Bob Hulsebosch, Bas van Schoonhoven, **Maya Sappelli**, Kees Wouters (2012) *User privacy in well-being and well-working applications: Requirements and approaches for user controlled privacy*, Proceedings of the Second International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT 2012).

21. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2012) *Using file system content to organize e-mail*, Proceedings of the fourth symposium on Information interaction in context (IIiX 2012)).

22. **Maya Sappelli**, Suzan Verberne, Wessel Kraaij (2012) *Supervision of learning methods in user data interpretation*, Proceedings of the fourth symposium on Information interaction in context (IIiX 2012).

23. **Maya Sappelli**, Suzan Verberne, Maarten van der Heijden, Max Hinne, Wessel Kraaij (2012) *Collecting ground truth data for query intent.*, Proceedings of the Dutch-Belgium Information Retrieval workshop (DIR 2012).

SIKS DISSERTATIONS

2009

2009-01 Rasa Jurgelenaite (RUN), *Symmetric Causal Independence Models.*
2009-02 Willem Robert van Hage (VU), *Evaluating Ontology-Alignment Techniques.*
2009-03 Hans Stol (UvT), *A Framework for Evidence-based Policy Making Using IT.*
2009-04 Josephine Nabukenya (RUN), *Improving the Quality of Organisational Policy Making using Collaboration Engineering.*
2009-05 Sietse Overbeek (RUN), *Bridging Supply and Demand for Knowledge Intensive Tasks - Based on Knowledge, Cognition, and Quality.*
2009-06 Muhammad Subianto (UU), *Understanding Classification.*
2009-07 Ronald Poppe (UT), *Discriminative Vision-Based Recovery and Recognition of Human Motion.*
2009-08 Volker Nannen (VU), *Evolutionary Agent-Based Policy Analysis in Dynamic Environments.*
2009-09 Benjamin Kanagwa (RUN), *Design, Discovery and Construction of Service-oriented Systems.*
2009-10 Jan Wielemaker (UVA), *Logic programming for knowledge-intensive interactive applications.*
2009-11 Alexander Boer (UVA), *Legal Theory, Sources of Law & the Semantic Web.*
2009-12 Peter Massuthe (TUE, Humboldt-Universitaet zu Berlin), *Operating Guidelines for Services.*
2009-13 Steven de Jong (UM), *Fairness in Multi-Agent Systems.*
2009-14 Maksym Korotkiy (VU), *From ontology-enabled services to service-enabled ontologies (making ontologies work in e-science with ONTO-SOA).*
2009-15 Rinke Hoekstra (UVA), *Ontology Representation - Design Patterns and Ontologies that Make Sense.*
2009-16 Fritz Reul (UvT), *New Architectures in Computer Chess.*
2009-17 Laurens van der Maaten (UvT), *Feature Extraction from Visual Data.*
2009-18 Fabian Groffen (CWI), *Armada, An Evolving Database System.*
2009-19 Valentin Robu (CWI), *Modeling Preferences, Strategic Reasoning and Collaboration in Agent-Mediated Electronic Markets.*
2009-20 Bob van der Vecht (UU), *Adjustable Autonomy: Controling Influences on Decision Making.*
2009-21 Stijn Vanderlooy (UM), *Ranking and Reliable Classification.*
2009-22 Pavel Serdyukov (UT), *Search For Expertise: Going beyond direct evidence.*
2009-23 Peter Hofgesang (VU), *Modelling Web Usage in a Changing Environment.*
2009-24 Annerieke Heuvelink (VUA), *Cognitive Models for Training Simulations.*
2009-25 Alex van Ballegooij (CWI), *"RAM: Array Database Management through Relational Mapping".*
2009-26 Fernando Koch (UU), *An Agent-Based Model for the Development of Intelligent Mobile Services.*
2009-27 Christian Glahn (OU), *Contextual Support of social Engagement and Reflection on the Web.*
2009-28 Sander Evers (UT), *Sensor Data Management with Probabilistic Models.*
2009-29 Stanislav Pokraev (UT), *Model-Driven Semantic Integration of Service-Oriented Applications.*
2009-30 Marcin Zukowski (CWI), *Balancing vectorized query execution with bandwidth-optimized storage.*
2009-31 Sofiya Katrenko (UVA), *A Closer Look at Learning Relations from Text.*
2009-32 Rik Farenhorst (VU) and Remco de Boer (VU), *Architectural Knowledge Management: Supporting Architects and Auditors.*
2009-33 Khiet Truong (UT), *How Does Real Affect Affect Affect Recognition In Speech?.*
2009-34 Inge van de Weerd (UU), *Advancing in Software Product Management: An Incremental Method Engineering Approach.*
2009-35 Wouter Koelewijn (UL), *Privacy en Politiegegevens; Over geautomatiseerde normatieve informatie-uitwisseling.*
2009-36 Marco Kalz (OUN), *Placement Support for Learners in Learning Networks.*
2009-37 Hendrik Drachsler (OUN), *Navigation Support for Learners in Informal Learning Networks.*
2009-38 Riina Vuorikari (OU), *Tags and self-organisation: a metadata ecology for learning resources in a multilingual context.*

2009-39 Christian Stahl (TUE, Humboldt-Universitaet zu Berlin), *Service Substitution – A Behavioral Approach Based on Petri Nets.*
2009-40 Stephan Raaijmakers (UvT), *Multinomial Language Learning: Investigations into the Geometry of Language.*
2009-41 Igor Berezhnyy (UvT), *Digital Analysis of Paintings.*
2009-42 Toine Bogers (UvT), *Recommender Systems for Social Bookmarking.*
2009-43 Virginia Nunes Leal Franqueira (UT), *Finding Multi-step Attacks in Computer Networks using Heuristic Search and Mobile Ambients.*
2009-44 Roberto Santana Tapia (UT), *Assessing Business-IT Alignment in Networked Organizations.*
2009-45 Jilles Vreeken (UU), *Making Pattern Mining Useful.*
2009-46 Loredana Afanasiev (UvA), *Querying XML: Benchmarks and Recursion.*

2010

2010-01 Matthijs van Leeuwen (UU), *Patterns that Matter.*
2010-02 Ingo Wassink (UT), *Work flows in Life Science.*
2010-03 Joost Geurts (CWI), *A Document Engineering Model and Processing Framework for Multimedia documents.*
2010-04 Olga Kulyk (UT), *Do You Know What I Know? Situational Awareness of Co-located Teams in Multidisplay Environments.*
2010-05 Claudia Hauff (UT), *Predicting the Effectiveness of Queries and Retrieval Systems.*
2010-06 Sander Bakkes (UvT), *Rapid Adaptation of Video Game AI.*
2010-07 Wim Fikkert (UT), *Gesture interaction at a Distance.*
2010-08 Krzysztof Siewicz (UL), *Towards an Improved Regulatory Framework of Free Software. Protecting user freedoms in a world of software communities and eGovernments.*
2010-09 Hugo Kielman (UL), *A Politiele gegevensverwerking en Privacy, Naar een effectieve waarborging.*
2010-10 Rebecca Ong (UL), *Mobile Communication and Protection of Children.*
2010-11 Adriaan Ter Mors (TUD), *The world according to MARP: Multi-Agent Route Planning.*
2010-12 Susan van den Braak (UU), *Sensemaking software for crime analysis.*
2010-13 Gianluigi Folino (RUN), *High Performance Data Mining using Bio-inspired techniques.*
2010-14 Sander van Splunter (VU), *Automated Web Service Reconfiguration.*
2010-15 Lianne Bodenstaff (UT), *Managing Dependency Relations in Inter-Organizational Models.*
2010-16 Sicco Verwer (TUD), *Efficient Identification of Timed Automata, theory and practice.*
2010-17 Spyros Kotoulas (VU), *Scalable Discovery of Networked Resources: Algorithms, Infrastructure, Applications.*
2010-18 Charlotte Gerritsen (VU), *Caught in the Act: Investigating Crime by Agent-Based Simulation.*
2010-19 Henriette Cramer (UvA), *People's Responses to Autonomous and Adaptive Systems.*
2010-20 Ivo Swartjes (UT), *Whose Story Is It Anyway? How Improv Informs Agency and Authorship of Emergent Narrative.*
2010-21 Harold van Heerde (UT), *Privacy-aware data management by means of data degradation.*
2010-22 Michiel Hildebrand (CWI), *End-user Support for Access to Heterogeneous Linked Data.*
2010-23 Bas Steunebrink (UU), *The Logical Structure of Emotions.*
2010-24 Dmytro Tykhonov, *Designing Generic and Efficient Negotiation Strategies.*
2010-25 Zulfiqar Ali Memon (VU), *Modelling Human-Awareness for Ambient Agents: A Human Mindreading Perspective.*
2010-26 Ying Zhang (CWI), *XRPC: Efficient Distributed Query Processing on Heterogeneous XQuery Engines.*
2010-27 Marten Voulon (UL), *Automatisch contracteren.*
2010-28 Arne Koopman (UU), *Characteristic Relational Patterns.*
2010-29 Stratos Idreos(CWI), *Database Cracking: Towards Auto-tuning Database Kernels.*
2010-30 Marieke van Erp (UvT), *Accessing Natural History - Discoveries in data cleaning, structuring, and retrieval.*
2010-31 Victor de Boer (UVA), *Ontology Enrichment from Heterogeneous Sources on the Web.*
2010-32 Marcel Hiel (UvT), *An Adaptive Service Oriented Architecture: Automatically solving Interoperability Problems.*
2010-33 Robin Aly (UT), *Modeling Representation Uncertainty in Concept-Based Multimedia Retrieval.*

2010-34 Teduh Dirgahayu (UT), *Interaction Design in Service Compositions.*
2010-35 Dolf Trieschnigg (UT), *Proof of Concept: Concept-based Biomedical Information Retrieval.*
2010-36 Jose Janssen (OU), *Paving the Way for Lifelong Learning; Facilitating competence development through a learning path specification.*
2010-37 Niels Lohmann (TUE), *Correctness of services and their composition.*
2010-38 Dirk Fahland (TUE), *From Scenarios to components.*
2010-39 Ghazanfar Farooq Siddiqui (VU), *Integrative modeling of emotions in virtual agents.*
2010-40 Mark van Assem (VU), *Converting and Integrating Vocabularies for the Semantic Web.*
2010-41 Guillaume Chaslot (UM), *Monte-Carlo Tree Search.*
2010-42 Sybren de Kinderen (VU), *Needs-driven service bundling in a multi-supplier setting - the computational e3-service approach.*
2010-43 Peter van Kranenburg (UU), *A Computational Approach to Content-Based Retrieval of Folk Song Melodies.*
2010-44 Pieter Bellekens (TUE), *An Approach towards Context-sensitive and User-adapted Access to Heterogeneous Data Sources, Illustrated in the Television Domain.*
2010-45 Vasilios Andrikopoulos (UvT), *A theory and model for the evolution of software services.*
2010-46 Vincent Pijpers (VU), *e3alignment: Exploring Inter-Organizational Business-ICT Alignment.*
2010-47 Chen Li (UT), *Mining Process Model Variants: Challenges, Techniques, Examples.*
2010-48 Withdrawn, .
2010-49 Jahn-Takeshi Saito (UM), *Solving difficult game positions.*
2010-50 Bouke Huurnink (UVA), *Search in Audiovisual Broadcast Archives.*
2010-51 Alia Khairia Amin (CWI), *Understanding and supporting information seeking tasks in multiple sources.*
2010-52 Peter-Paul van Maanen (VU), *Adaptive Support for Human-Computer Teams: Exploring the Use of Cognitive Models of Trust and Attention.*
2010-53 Edgar Meij (UVA), *Combining Concepts and Language Models for Information Access.*

2011
2011-01 Botond Cseke (RUN), *Variational Algorithms for Bayesian Inference in Latent Gaussian Models.*
2011-02 Nick Tinnemeier(UU), *Organizing Agent Organizations. Syntax and Operational Semantics of an Organization-Oriented Programming Language.*
2011-03 Jan Martijn van der Werf (TUE), *Compositional Design and Verification of Component-Based Information Systems.*
2011-04 Hado van Hasselt (UU), *Insights in Reinforcement Learning; Formal analysis and empirical evaluation of temporal-difference.*
2011-05 Base van der Raadt (VU), *Enterprise Architecture Coming of Age - Increasing the Performance of an Emerging Discipline..*
2011-06 Yiwen Wang (TUE), *Semantically-Enhanced Recommendations in Cultural Heritage.*
2011-07 Yujia Cao (UT), *Multimodal Information Presentation for High Load Human Computer Interaction.*
2011-08 Nieske Vergunst (UU), *BDI-based Generation of Robust Task-Oriented Dialogues.*
2011-09 Tim de Jong (OU), *Contextualised Mobile Media for Learning.*
2011-10 Bart Bogaert (UvT), *Cloud Content Contention.*
2011-11 Dhaval Vyas (UT), *Designing for Awareness: An Experience-focused HCI Perspective.*
2011-12 Carmen Bratosin (TUE), *Grid Architecture for Distributed Process Mining.*
2011-13 Xiaoyu Mao (UvT), *Airport under Control. Multiagent Scheduling for Airport Ground Handling.*
2011-14 Milan Lovric (EUR), *Behavioral Finance and Agent-Based Artificial Markets.*
2011-15 Marijn Koolen (UvA), *The Meaning of Structure: the Value of Link Evidence for Information Retrieval.*
2011-16 Maarten Schadd (UM), *Selective Search in Games of Different Complexity.*
2011-17 Jiyin He (UVA), *Exploring Topic Structure: Coherence, Diversity and Relatedness.*
2011-18 Mark Ponsen (UM), *Strategic Decision-Making in complex games.*
2011-19 Ellen Rusman (OU), *The Mind ' s Eye on Personal Profiles.*
2011-20 Qing Gu (VU), *Guiding service-oriented software engineering - A view-based approach.*
2011-21 Linda Terlouw (TUD), *Modularization and Specification of Service-Oriented Systems.*
2011-22 Junte Zhang (UVA), *System Evaluation of Archival Description and Access.*

2011-23 Wouter Weerkamp (UVA), *Finding People and their Utterances in Social Media.*
2011-24 Herwin van Welbergen (UT), *Behavior Generation for Interpersonal Coordination with Virtual Humans On Specifying, Scheduling and Realizing Multimodal Virtual Human Behavior.*
2011-25 Syed Waqar ul Qounain Jaffry (VU)), *Analysis and Validation of Models for Trust Dynamics.*
2011-26 Matthijs Aart Pontier (VU), *Virtual Agents for Human Communication - Emotion Regulation and Involvement-Distance Trade-Offs in Embodied Conversational Agents and Robots.*
2011-27 Aniel Bhulai (VU), *Dynamic website optimization through autonomous management of design patterns.*
2011-28 Rianne Kaptein(UVA), *Effective Focused Retrieval by Exploiting Query Context and Document Structure.*
2011-29 Faisal Kamiran (TUE), *Discrimination-aware Classification.*
2011-30 Egon van den Broek (UT), *Affective Signal Processing (ASP): Unraveling the mystery of emotions.*
2011-31 Ludo Waltman (EUR), *Computational and Game-Theoretic Approaches for Modeling Bounded Rationality.*
2011-32 Nees-Jan van Eck (EUR), *Methodological Advances in Bibliometric Mapping of Science.*
2011-33 Tom van der Weide (UU), *Arguing to Motivate Decisions.*
2011-34 Paolo Turrini (UU), *Strategic Reasoning in Interdependence: Logical and Game-theoretical Investigations.*
2011-35 Maaike Harbers (UU), *Explaining Agent Behavior in Virtual Training.*
2011-36 Erik van der Spek (UU), *Experiments in serious game design: a cognitive approach.*
2011-37 Adriana Burlutiu (RUN), *Machine Learning for Pairwise Data, Applications for Preference Learning and Supervised Network Inference.*
2011-38 Nyree Lemmens (UM), *Bee-inspired Distributed Optimization.*
2011-39 Joost Westra (UU), *Organizing Adaptation using Agents in Serious Games.*
2011-40 Viktor Clerc (VU), *Architectural Knowledge Management in Global Software Development.*
2011-41 Luan Ibraimi (UT), *Cryptographically Enforced Distributed Data Access Control.*
2011-42 Michal Sindlar (UU), *Explaining Behavior through Mental State Attribution.*
2011-43 Henk van der Schuur (UU), *Process Improvement through Software Operation Knowledge.*
2011-44 Boris Reuderink (UT), *Robust Brain-Computer Interfaces.*
2011-45 Herman Stehouwer (UvT), *Statistical Language Models for Alternative Sequence Selection.*
2011-46 Beibei Hu (TUD), *Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work.*
2011-47 Azizi Bin Ab Aziz(VU), *Exploring Computational Models for Intelligent Support of Persons with Depression.*
2011-48 Mark Ter Maat (UT), *Response Selection and Turn-taking for a Sensitive Artificial Listening Agent.*
2011-49 Andreea Niculescu (UT), *Conversational interfaces for task-oriented spoken dialogues: design aspects influencing interaction quality.*

2012

2012-01 Terry Kakeeto (UvT), *Relationship Marketing for SMEs in Uganda.*
2012-02 Muhammad Umair(VU), *Adaptivity, emotion, and Rationality in Human and Ambient Agent Models.*
2012-03 Adam Vanya (VU), *Supporting Architecture Evolution by Mining Software Repositories.*
2012-04 Jurriaan Souer (UU), *Development of Content Management System-based Web Applications.*
2012-05 Marijn Plomp (UU), *Maturing Interorganisational Information Systems.*
2012-06 Wolfgang Reinhardt (OU), *Awareness Support for Knowledge Workers in Research Networks.*
2012-07 Rianne van Lambalgen (VU), *When the Going Gets Tough: Exploring Agent-based Models of Human Performance under Demanding Conditions.*
2012-08 Gerben de Vries (UVA), *Kernel Methods for Vessel Trajectories.*
2012-09 Ricardo Neisse (UT), *Trust and Privacy Management Support for Context-Aware Service Platforms.*
2012-10 David Smits (TUE), *Towards a Generic Distributed Adaptive Hypermedia Environment.*
2012-11 J.C.B. Rantham Prabhakara (TUE), *Process Mining in the Large: Preprocessing, Discovery, and Diagnostics.*
2012-12 Kees van der Sluijs (TUE), *Model Driven Design and Data Integration in Semantic Web Information Systems.*

2012-13 Suleman Shahid (UvT), *Fun and Face: Exploring non-verbal expressions of emotion during playful interactions.*
2012-14 Evgeny Knutov(TUE), *Generic Adaptation Framework for Unifying Adaptive Web-based Systems.*
2012-15 Natalie van der Wal (VU), *Social Agents. Agent-Based Modelling of Integrated Internal and Social Dynamics of Cognitive and Affective Processes..*
2012-16 Fiemke Both (VU), *Helping people by understanding them - Ambient Agents supporting task execution and depression treatment.*
2012-17 Amal Elgammal (UvT), *Towards a Comprehensive Framework for Business Process Compliance.*
2012-18 Eltjo Poort (VU), *Improving Solution Architecting Practices.*
2012-19 Helen Schonenberg (TUE), *What's Next? Operational Support for Business Process Execution.*
2012-20 Ali Bahramisharif (RUN), *Covert Visual Spatial Attention, a Robust Paradigm for Brain-Computer Interfacing.*
2012-21 Roberto Cornacchia (TUD), *Querying Sparse Matrices for Information Retrieval.*
2012-22 Thijs Vis (UvT), *Intelligence, politie en veiligheidsdienst: verenigbare grootheden?.*
2012-23 Christian Muehl (UT), *Toward Affective Brain-Computer Interfaces: Exploring the Neurophysiology of Affect during Human Media Interaction.*
2012-24 Laurens van der Werff (UT), *Evaluation of Noisy Transcripts for Spoken Document Retrieval.*
2012-25 Silja Eckartz (UT), *Managing the Business Case Development in Inter-Organizational IT Projects: A Methodology and its Application.*
2012-26 Emile de Maat (UVA), *Making Sense of Legal Text.*
2012-27 Hayrettin Gurkok (UT), *Mind the Sheep! User Experience Evaluation & Brain-Computer Interface Games.*
2012-28 Nancy Pascall (UvT), *Engendering Technology Empowering Women.*
2012-29 Almer Tigelaar (UT), *Peer-to-Peer Information Retrieval.*
2012-30 Alina Pommeranz (TUD), *Designing Human-Centered Systems for Reflective Decision Making.*
2012-31 Emily Bagarukayo (RUN), *A Learning by Construction Approach for Higher Order Cognitive Skills Improvement, Building Capacity and Infrastructure.*
2012-32 Wietske Visser (TUD), *Qualitative multi-criteria preference representation and reasoning.*
2012-33 Rory Sie (OUN), *Coalitions in Cooperation Networks (COCOON).*
2012-34 Pavol Jancura (RUN), *Evolutionary analysis in PPI networks and applications.*
2012-35 Evert Haasdijk (VU), *Never Too Old To Learn – On-line Evolution of Controllers in Swarm- and Modular Robotics.*
2012-36 Denis Ssebugwawo (RUN), *Analysis and Evaluation of Collaborative Modeling Processes.*
2012-37 Agnes Nakakawa (RUN), *A Collaboration Process for Enterprise Architecture Creation.*
2012-38 Selmar Smit (VU), *Parameter Tuning and Scientific Testing in Evolutionary Algorithms.*
2012-39 Hassan Fatemi (UT), *Risk-aware design of value and coordination networks.*
2012-40 Agus Gunawan (UvT), *Information Access for SMEs in Indonesia.*
2012-41 Sebastian Kelle (OU), *Game Design Patterns for Learning.*
2012-42 Dominique Verpoorten (OU), *Reflection Amplifiers in self-regulated Learning.*
2012-43 Withdrawn, .
2012-44 Anna Tordai (VU), *On Combining Alignment Techniques.*
2012-45 Benedikt Kratz (UvT), *A Model and Language for Business-aware Transactions.*
2012-46 Simon Carter (UVA), *Exploration and Exploitation of Multilingual Data for Statistical Machine Translation.*
2012-47 Manos Tsagkias (UVA), *Mining Social Media: Tracking Content and Predicting Behavior.*
2012-48 Jorn Bakker (TUE), *Handling Abrupt Changes in Evolving Time-series Data.*
2012-49 Michael Kaisers (UM), *Learning against Learning - Evolutionary dynamics of reinforcement learning algorithms in strategic interactions.*
2012-50 Steven van Kervel (TUD), *Ontologogy driven Enterprise Information Systems Engineering.*
2012-51 Jeroen de Jong (TUD), *Heuristics in Dynamic Sceduling; a practical framework with a case study in elevator dispatching.*

2013

2013-01 Viorel Milea (EUR), *News Analytics for Financial Decision Support.*
2013-02 Erietta Liarou (CWI), *MonetDB/DataCell: Leveraging the Column-store Database Technology for Efficient and Scalable Stream Processing.*

2013-03 Szymon Klarman (VU), *Reasoning with Contexts in Description Logics.*
2013-04 Chetan Yadati(TUD), *Coordinating autonomous planning and scheduling.*
2013-05 Dulce Pumareja (UT), *Groupware Requirements Evolutions Patterns.*
2013-06 Romulo Goncalves(CWI), *The Data Cyclotron: Juggling Data and Queries for a Data Warehouse Audience.*
2013-07 Giel van Lankveld (UvT), *Quantifying Individual Player Differences.*
2013-08 Robbert-Jan Merk(VU), *Making enemies: cognitive modeling for opponent agents in fighter pilot simulators.*
2013-09 Fabio Gori (RUN), *Metagenomic Data Analysis: Computational Methods and Applications.*
2013-10 Jeewanie Jayasinghe Arachchige(UvT), *A Unified Modeling Framework for Service Design..*
2013-11 Evangelos Pournaras(TUD), *Multi-level Reconfigurable Self-organization in Overlay Services.*
2013-12 Marian Razavian(VU), *Knowledge-driven Migration to Services.*
2013-13 Mohammad Safiri(UT), *Service Tailoring: User-centric creation of integrated IT-based homecare services to support independent living of elderly.*
2013-14 Jafar Tanha (UVA), *Ensemble Approaches to Semi-Supervised Learning Learning.*
2013-15 Daniel Hennes (UM), *Multiagent Learning - Dynamic Games and Applications.*
2013-16 Eric Kok (UU), *Exploring the practical benefits of argumentation in multi-agent deliberation.*
2013-17 Koen Kok (VU), *The PowerMatcher: Smart Coordination for the Smart Electricity Grid.*
2013-18 Jeroen Janssens (UvT), *Outlier Selection and One-Class Classification.*
2013-19 Renze Steenhuizen (TUD), *Coordinated Multi-Agent Planning and Scheduling.*
2013-20 Katja Hofmann (UvA), *Fast and Reliable Online Learning to Rank for Information Retrieval.*
2013-21 Sander Wubben (UvT), *Text-to-text generation by monolingual machine translation.*
2013-22 Tom Claassen (RUN), *Causal Discovery and Logic.*
2013-23 Patricio de Alencar Silva(UvT), *Value Activity Monitoring.*
2013-24 Haitham Bou Ammar (UM), *Automated Transfer in Reinforcement Learning.*
2013-25 Agnieszka Anna Latoszek-Berendsen (UM), *Intention-based Decision Support. A new way of representing and implementing clinical guidelines in a Decision Support System.*
2013-26 Alireza Zarghami (UT), *Architectural Support for Dynamic Homecare Service Provisioning.*
2013-27 Mohammad Huq (UT), *Inference-based Framework Managing Data Provenance.*
2013-28 Frans van der Sluis (UT), *When Complexity becomes Interesting: An Inquiry into the Information eXperience.*
2013-29 Iwan de Kok (UT), *Listening Heads.*
2013-30 Joyce Nakatumba (TUE), *Resource-Aware Business Process Management: Analysis and Support.*
2013-31 Dinh Khoa Nguyen (UvT), *Blueprint Model and Language for Engineering Cloud Applications.*
2013-32 Kamakshi Rajagopal (OUN), *Networking For Learning; The role of Networking in a Lifelong Learner's Professional Development.*
2013-33 Qi Gao (TUD), *User Modeling and Personalization in the Microblogging Sphere.*
2013-34 Kien Tjin-Kam-Jet (UT), *Distributed Deep Web Search.*
2013-35 Abdallah El Ali (UvA), *Minimal Mobile Human Computer Interaction.*
2013-36 Than Lam Hoang (TUe), *Pattern Mining in Data Streams.*
2013-37 Dirk Börner (OUN), *Ambient Learning Displays.*
2013-38 Eelco den Heijer (VU), *Autonomous Evolutionary Art.*
2013-39 Joop de Jong (TUD), *A Method for Enterprise Ontology based Design of Enterprise Information Systems.*
2013-40 Pim Nijssen (UM), *Monte-Carlo Tree Search for Multi-Player Games.*
2013-41 Jochem Liem (UVA), *Supporting the Conceptual Modelling of Dynamic Systems: A Knowledge Engineering Perspective on Qualitative Reasoning.*
2013-42 Léon Planken (TUD), *Algorithms for Simple Temporal Reasoning.*
2013-43 Marc Bron (UVA), *Exploration and Contextualization through Interaction and Concepts.*

2014

2014-01 Nicola Barile (UU), *Studies in Learning Monotone Models from Data.*
2014-02 Fiona Tuliyano (RUN), *Combining System Dynamics with a Domain Modeling Method.*
2014-03 Sergio Raul Duarte Torres (UT), *Information Retrieval for Children: Search Behavior and Solutions.*
2014-04 Hanna Jochmann-Mannak (UT), *Websites for children: search strategies and interface design -*

Three studies on children's search performance and evaluation.
2014-05 Jurriaan van Reijsen (UU), *Knowledge Perspectives on Advancing Dynamic Capability.*
2014-06 Damian Tamburri (VU), *Supporting Networked Software Development.*
2014-07 Arya Adriansyah (TUE), *Aligning Observed and Modeled Behavior.*
2014-08 Samur Araujo (TUD), *Data Integration over Distributed and Heterogeneous Data Endpoints.*
2014-09 Philip Jackson (UvT), *Toward Human-Level Artificial Intelligence: Representation and Computation of Meaning in Natural Language.*
2014-10 Ivan Salvador Razo Zapata (VU), *Service Value Networks.*
2014-11 Janneke van der Zwaan (TUD), *An Empathic Virtual Buddy for Social Support.*
2014-12 Willem van Willigen (VU), *Look Ma, No Hands: Aspects of Autonomous Vehicle Control.*
2014-13 Arlette van Wissen (VU), *Agent-Based Support for Behavior Change: Models and Applications in Health and Safety Domains.*
2014-14 Yangyang Shi (TUD), *Language Models With Meta-information.*
2014-15 Natalya Mogles (VU), *Agent-Based Analysis and Support of Human Functioning in Complex Socio-Technical Systems: Applications in Safety and Healthcare.*
2014-16 Krystyna Milian (VU), *Supporting trial recruitment and design by automatically interpreting eligibility criteria.*
2014-17 Kathrin Dentler (VU), *Computing healthcare quality indicators automatically: Secondary Use of Patient Data and Semantic Interoperability.*
2014-18 Mattijs Ghijsen (VU), *Methods and Models for the Design and Study of Dynamic Agent Organizations.*
2014-19 Vincius Ramos (TUE), *Adaptive Hypermedia Courses: Qualitative and Quantitative Evaluation and Tool Support.*
2014-20 Mena Habib (UT), *Named Entity Extraction and Disambiguation for Informal Text: The Missing Link.*
2014-21 Kassidy Clark (TUD), *Negotiation and Monitoring in Open Environments.*
2014-22 Marieke Peeters (UU), *Personalized Educational Games - Developing agent-supported scenario-based training.*
2014-23 Eleftherios Sidirourgos (UvA/CWI), *Space Efficient Indexes for the Big Data Era.*
2014-24 Davide Ceolin (VU), *Trusting Semi-structured Web Data.*
2014-25 Martijn Lappenschaar (RUN), *New network models for the analysis of disease interaction.*
2014-26 Tim Baarslag (TUD), *What to Bid and When to Stop.*
2014-27 Rui Jorge Almeida (EUR), *Conditional Density Models Integrating Fuzzy and Probabilistic Representations of Uncertainty.*
2014-28 Anna Chmielowiec (VU), *Decentralized k-Clique Matching.*
2014-29 Jaap Kabbedijk (UU), *Variability in Multi-Tenant Enterprise Software.*
2014-30 Peter de Cock (UvT), *Anticipating Criminal Behaviour.*
2014-31 Leo van Moergestel (UU), *Agent Technology in Agile Multiparallel Manufacturing and Product Support.*
2014-32 Naser Ayat (UvA), *On Entity Resolution in Probabilistic Data.*
2014-33 Tesfa Tegegne (RUN), *Service Discovery in eHealth.*
2014-34 Christina Manteli(VU), *The Effect of Governance in Global Software Development: Analyzing Transactive Memory Systems..*
2014-35 Joost van Ooijen (UU), *Cognitive Agents in Virtual Worlds: A Middleware Design Approach.*
2014-36 Joos Buijs (TUE), *Flexible Evolutionary Algorithms for Mining Structured Process Models.*
2014-37 Maral Dadvar (UT), *Experts and Machines United Against Cyberbullying.*
2014-38 Danny Plass-Oude Bos (UT), *Making brain-computer interfaces better: improving usability through post-processing..*
2014-39 Jasmina Maric (UvT), *Web Communities, Immigration, and Social Capital.*
2014-40 Walter Omona (RUN), *A Framework for Knowledge Management Using ICT in Higher Education.*
2014-41 Frederic Hogenboom (EUR), *Automated Detection of Financial Events in News Text.*
2014-42 Carsten Eijckhof (CWI/TUD), *Contextual Multidimensional Relevance Models.*
2014-43 Kevin Vlaanderen (UU), *Supporting Process Improvement using Method Increments.*
2014-44 Paulien Meesters (UvT), *Intelligent Blauw. Met als ondertitel: Intelligence-gestuurde politiezorg in gebiedsgebonden eenheden..*
2014-45 Birgit Schmitz (OUN), *Mobile Games for Learning: A Pattern-Based Approach.*
2014-46 Ke Tao (TUD), *Social Web Data Analytics: Relevance, Redundancy, Diversity.*

2014-47 Shangsong Liang (UVA), *Fusion and Diversification in Information Retrieval.*

2015

2015-01 Niels Netten (UvA), *Machine Learning for Relevance of Information in Crisis Response.*
2015-02 Faiza Bukhsh (UvT), *Smart auditing: Innovative Compliance Checking in Customs Controls.*
2015-03 Twan van Laarhoven (RUN), *Machine learning for network data.*
2015-04 Howard Spoelstra (OUN), *Collaborations in Open Learning Environments.*
2015-05 Christoph Bösch(UT), *Cryptographically Enforced Search Pattern Hiding.*
2015-06 Farideh Heidari (TUD), *Business Process Quality Computation - Computing Non-Functional Requirements to Improve Business Processes.*
2015-07 Maria-Hendrike Peetz(UvA), *Time-Aware Online Reputation Analysis.*
2015-08 Jie Jiang (TUD), *Organizational Compliance: An agent-based model for designing and evaluating organizational interactions.*
2015-09 Randy Klaassen(UT), *HCI Perspectives on Behavior Change Support Systems.*
2015-10 Henry Hermans (OUN), *OpenU: design of an integrated system to support lifelong learning.*
2015-11 Yongming Luo (TUE), *Designing algorithms for big graph datasets: A study of computing bisimulation and joins.*
2015-12 Julie M. Birkholz (VU), *Modi Operandi of Social Network Dynamics: The Effect of Context on Scientific Collaboration Networks.*
2015-13 Giuseppe Procaccianti(VU), *Energy-Efficient Software.*
2015-14 Bart van Straalen (UT), *A cognitive approach to modeling bad news conversations.*
2015-15 Klaas Andries de Graaf (VU), *Ontology-based Software Architecture Documentation.*
2015-16 Changyun Wei (UT), *Cognitive Coordination for Cooperative Multi-Robot Teamwork.*
2015-17 Andr? van Cleeff (UT), *Physical and Digital Security Mechanisms: Properties, Combinations and Trade-offs.*
2015-18 Holger Pirk (CWI), *Waste Not, Want Not! - Managing Relational Data in Asymmetric Memories.*
2015-19 Bernardo Tabuenca (OUN), *Ubiquitous Technology for Lifelong Learners.*
2015-20 Lo?s Vanh?e(UU), *Using Culture and Values to Support Flexible Coordination.*
2015-21 Sibren Fetter (OUN), *Using Peer-Support to Expand and Stabilize Online Learning.*
2015-23 Luit Gazendam (VU), *Cataloguer Support in Cultural Heritage.*
2015-24 Richard Berendsen (UVA), *Finding People, Papers, and Posts: Vertical Search Algorithms and Evaluation.*
2015-25 Steven Woudenberg (UU), *Bayesian Tools for Early Disease Detection.*
2015-26 Alexander Hogenboom (EUR), *Sentiment Analysis of Text Guided by Semantics and Structure.*
2015-27 Sándor Héman (CWI), *Updating compressed column-stores.*
2015-28 Janet Bagorogoza (TiU), *Knowledge Management and High Performance; The Uganda Financial Institutions Model for HPO.*
2015-29 Hendrik Baier (UM), *Monte-Carlo Tree Search Enhancements for One-Player and Two-Player Domains.*
2015-30 Kiavash Bahreini (OUN), *Real-time Multimodal Emotion Recognition in E-Learning.*
2015-31 Yakup Koç (TUD), *On Robustness of Power Grids.*
2015-32 Jerome Gard (UL), *Corporate Venture Management in SMEs.*
2015-33 Frederik Schadd (UM), *Ontology Mapping with Auxiliary Resources.*
2015-34 Victor de Graaff (UT), *Geosocial Recommender Systems.*
2015-35 Junchao Xu (TUD), *Affective Body Language of Humanoid Robots: Perception and Effects in Human Robot Interaction.*

2016

2016-01 Syed Saiden Abbas (RUN), *Recognition of Shapes by Humans and Machines.*
2016-02 Michiel Christiaan Meulendijk (UU), *Optimizing medication reviews through decision support: prescribing a better pill to swallow.*
2016-03 Maya Sappelli (RUN), *Knowledge Work in Context: User Centered Knowledge Worker Support.*