Complex Networks Analysis

A DISSERTATION PRESENTED
BY
ALEXANDRU-PETRUȚ BECHERU
ADVISED BY
PROF. UNIV. DR. ING. COSTIN BĂDICĂ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN THE SUBJECT OF
COMPUTERS & INFORMATION TECHNOLOGY

UNIVERSITY OF CRAIOVA
CRAIOVA, ROMANIA
SEPTEMBER 2019

Advisor: Prof. univ. dr. ing. Costin Bădică Alexandru-Petruț Becheru

Complex Networks Analysis

Abstract

The current understanding of the surrounding world shows us that nature is formed out of complex interconnecting systems. Networks created by these systems support phenomena that are far from being deterministic trough traditional methods, or fully addressable by one research field. As in Edward Lorenz's *butterfly effect*, an element influences the network, while the network puts its mark on every element. Further more, the complexity is constantly increasing due to the technological advancement, large scale of data available, increased interaction and overlapping of various phenomena. Thus, the scope of this dissertation is to address and contribute to the current understanding of complex interconnected systems.

One of the fields of research that has been shown capable of modelling and evaluating such complex systems is *Complex Networks Analysis* (CNA), otherwise called *Network Science* (NS). This research field leverages knowledge from various other research domains such as: *Graph Theory, Computer Science, Physics, Social Sciences*; just to name those fields that contributed the most to CNA. One commonly accepted definition of this fields of research is that it investigates non-trivial real-world features, that are neither purely regular nor purely random, which that can be modelled as graphs. Hence, the tools and methods used in this dissertation are mainly CNA based.

As no dissertation can even hope to address all domains where complex systems arise, we shall focus on several heterogeneous domains, i.e. Freight Transportation, Tourism, E-learning and Organisational Analysis. In our quest to extract and provide knowledge in the domains of study we shall tackle with the fol-

lowing processes: data acquisition, data preprocessing, data storage, complex network creation (data modelling), complex networks analysis, concept proofing.

As part of the Freight Transportation domain this dissertation shall address the development and validation of a *multi agent framework for the brokering of freight transport*. Ontologies shall be developed to define the semantic modelling of information relevant to freight transport and provide means of communication and interaction among agents. Further more, we shall provide proof, through conducting experiments, that state of the art negotiation mechanisms, i.e. ICNET, can solve brokering related conflicts. Nevertheless, we shall introduce a novel CNA inspired negotiation mechanism through which complex dynamic business strategies can be implemented with success.

As touristic internet portals have become a major asset for the Tourism domain, we devoted our attention to achieving a better understating of collaborations that arise on a Romanian touristic internet portal, i.e. `http://amfostacolo.ro/`. Valuable insights on how collaborations arise and develop over time will be present as well as general traits of the community of users, e.g. its meritocratic nature. Furthermore, a novel CNA and natural language processing (NLP) based method for text summarising via keyword identification and extraction will be presented. This method will be proven to have a superior performance than the widely used *term frequency-inverse document frequency* NLP method. Nevertheless, we will provide evidence that CNA methods can be used to determine whether textual reviews were automatically generated by software bots through inspecting the cohesion of networks generated from reviews.

The adoption of new technologies in the learning process has facilitated the rise of *E-learning* and complex *Social Learning Environments*. Therefor, we have

conducted two studies that relate to one such environment used during a course of *Web Applications Development* at the University of Craiova. The first study provides evidence that textual complexity indices corroborated with the time evolution of learners posts can be used to predict their academic performance as well as their learning style. The second study use CNA methods to propose and validate a knowledge extraction framework in order to provide much needed pedagogical insights. Further more, we introduce an pedagogical inspired network visualisation platform for social learning environments, that facilitates the use of the proposed knowledge extraction framework by the average computer literate instructor.

Last but not least, we have used CNA as means of evaluating the knowledge exchange within organisations. A study on the SCRUM organisation model will quantify, for the first time, the advantages of using a SCRUM organisation model over the classical hierarchical model, from a knowledge exchange perspective. Next, we shall introduce an ontology based method for inferring the type of a complex network. Thus, we provide the necessary means through which insights on the stability, development, knowledge exchange and social phenomena of a complex network can be rapidly and easily obtained. A use case on this ontology is presented in the context of analysing organisational networks. Further more, a CNA based organisational analysis shall be conducted on an academical organisation. The analysis will provide valuable insights on how the knowledge exchange in such a context is done and provide further proof on the utility of CNA in organisational studies.

Contents

Figures

Tables

Equations

Listings

Abbreviations

AC RO-GER Academical Network

ACID Atomicity Consistency Isolation Durability

ACL Agent Communication Language

AD RO-GER Administrative Network

AICNET Augmented Iterated Contract Net protocol

BA Business Analyst

CNA Complex Network Analysis

CNET Contract Net protocol

CSCL Computer Supported Collaborative Learning

CSV Comma Separated values file

D Designer

DFA Discriminant Function Analysis

Dm Design Manager

Ds Team of Designers

DtoG Data acquisition & Graph building module in StudentViz

eMUSE empowering MashUps for Social E-learning platform

ES Experimental System

FCN RO-GER Full Communication Network

FIPA Foundation for Intelligent Physical Agents

FM RO-GER Mail Network for February-March

FormNet RO-GER Network computed as aggreate of AC, AD, FM and Social Networks

FSLSM Felder and Silverman Learning Style Model

ICNET Iterated Contract Net

ICT Information & Communication Technologies

ILS Index of Learning Styles questionnaire

MAFTLS Multi Agent Framework for Transport Logistics Service

MailNet RO-GER Mail Network aggregated from FM and MJ networks

MJ RO-GER Mail Network for May-June

NLP Natural Language Processing

ONA Organisational Network Analysis

OrgNet RO-GER Organisational Chart Network

P Programmer

PBL Project Based Learning

PM Project Manager

PRM Programming Manager

PTn Team of programmers

QA Quality Assurance employee

QAM Quality Assurance Manager

RB ReaderBench

RO-GER Romanian-German University of Sibiu, Romania

sCA AICNET strategy Competitiveness Advantage

SDM Software Developer Manager

SK Stakeholders

SLE Social Learning Environment

SNA Social Network Analysis

sPR AICNET strategy PageRank

tf-idf term frequency–inverse document frequency

WAD Web Applications Design course

I dedicate this dissertation to my wife, parents, friends and colleagues for their support in completing this endeavour.

I would like to thank all my teachers, with notable regards for:

Costin Badică

Constantin Cazacu

Valentin Cenuşe

Mihaela Codreş

Nicolae Constantinescu

Mirel Coşulschi

Venera Ifrim

Elvira Popescu

Mihaela Sterpu

Cătălin Stoean

Acknowledgments

I would like to present my regards to the Romanian-German University of Sibiu (Romania) and the internet portal http://amfostacolo.ro/ for providing and allowing me to use their data in research scopes. Also, I would like to acknowledge that my research was supported through academical/research grants as follows:

2013-2016 Grant Ph.d for studies on Complex Networks, University of Craiova.

2014-2015 Grant POS-DRU/159/1.5/2/133255. Project ID 133225(2014), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007-2013.

2015-2017 Grant of the Romanian National Authority for Scientific Research and Innovation, CNCS – UEFISCDI, project number PN-II-RU-TE-2014-4-2604.

2017-2018 Grant QFORIT, University of Craiova.

Contributing Authors

The following authors contributed to Chapter 1: Costin Bădică (Ph.D), Lucian Luncean (M.Sc) & Andrei Mocanu (M.Sc);
The following authors contributed to Chapter 2: Costin Bădică (Ph.D), Florin Buşe (M.Sc) & Mihaela Colhon (Ph.D);
The following authors contributed to Chapter 3: Andreea Calotă (M.Sc), Scott Andrew Crossley (Ph.D), Mihai Dascălu (Ph.D), Stefan Trausan-Matu (Ph.D) & Elvira Popescu (Ph.D);
The following authors contributed to Chapter 4: Costin Bădică (Ph.D);

A computer would deserve to be called intelligent if it could deceive a human into believing that it was human.

Alan Turing

1

Freight Transport Framework for Logistics Service

In recent years the development of Internet applications has known a considerable growth both in quality and quantity. Many entrepreneurs have identified an opportunity to increase their business. One of these opportunities is the constant struggle to automate operations in the domain of freight transport logistics.

In this chapter we shall discuss an agent-based system that fits within the business model of freight transportation exchanges. It can be defined as virtual logistics platform that operates in the domain of freight transport, where providers of freight, as well as cargo owners are continuously seeking new transport opportunities. The basic operating scenario of the system can be summarised as following. We consider a broker position in a transport company holding a Website where requests and offers are posted. Requests are posted by the cargo owners and offers are posted by the providers of freight. Both the transport provider and the cargo owner are continuously seeking transport opportunities. The role of the freight

transportation broker is to match requests with offers, such that goods are transported in optimal conditions. The broker has to take in consideration the time constraints, vehicle capabilities and to generate revenue for itself in order to remain operational.

A logistics service with brokering functionality is proposed based on an augmented *Complex Networks Analysis* agent-based negotiation, further mentioned as Multi Agent Freight Transport Framework for Logistics Service (MAFTLS). Upon receiving a transport request, a virtual supply chain emerges from the system through the automated negotiation processes between software agents. Through the use of CNA methods/techniques additional leverage/information will be provided tot the service; thus, allowing the service to use complex negotiation scenarios.

This chapter is structured as following. The following section will address background information and related work. The third section will discuss the semantic modelling of information relevant to freight transportation exchanges. The communication and interaction means of the proposed system will be introduced in the forth section. The fifth section will tackle the negotiation process, from a state of the art protocol (ICNET). In the sixth section we will introduce a novel negotiation protocol based on CNA methods. The last section contain our concluding remarks and identified future work.

1.1 Background & Related Work

1.1.1 Ontologies

Ontologies are a formal, shared and explicit declarative description of a domain. An ontology contains an hierarchically organised set of classes or concepts. Subclasses represent concepts that are more specific than their super classes. Each class contains a set of properties describing the features and attributes of the concept. Properties can be characterised by restrictions that constrain the values of the property. An ontology and a set of individual instances of classes represent a knowledge base. Ontologies are increasingly used in various fields [108] such as knowledge management, natural language processing, e-commerce, linguistics

[113], intelligent information integration, education; and in new emerging fields like the Semantic Web.

Good ontologies are those that serve their purpose. Complete ontologies are probably more than what most knowledge services really need to function properly. A possible correct answer to the question "*How do you know if your ontology is good?*" is "*If it works well in your application.*" According to this view, the ontology size, the complexity of the ontology and the vocabulary of the ontology do not really matter for the assessment of an ontology. Vrandečić D. has proposed in paper [224] the following three issues as relevant for ontology evaluation: (i) mistakes and omissions in ontologies can lead to the inability of applications to achieve the full potential of exchanged data, (ii) people constructing an ontology need a way to evaluate their results and possibly to guide the construction process, as well as any subsequent refinement steps and (iii) local changes in collaborative ontology engineering may affect the work of others. Paper [89] looks at existing ontology-evaluation methods from the perspective of their integration in one single framework.

The management of an innovative and modern supply chain (in our case transport services) requires the use of appropriate logistics performance. More about logistics performance can be seen in [217] where the authors indicate the relevant and important indicators and performance metrics for intelligent and sustainable supply chains. The authors of papers [71, 226] have modified and extend an existing ontology *CoreGRID* with the aim to facilitate ontological-driven resource management in the grid. Besides the proposed scenario and description of the developed *AiG* (Agents in the Grid) Ontology, the authors present two examples of resource descriptions in order to evaluate their proposed ontology. Moreover, the authors utilise the ontology to describe contract conditions as well as content of messages exchanged by software agents. Therefore, the purpose of the *AiG Ontology* has many similarities with our *Logistics Ontology*.

During our research we decided to adapt an ontology initially developed by the project InterLogGrid [109, 195] for our proposed system. The aforementioned ontology was used as a basic skeleton where we could add or remove concepts in order to better represent freight transport scenarios. However due to numerous and significant changes, the two ontologies are now substantially distinct.

The authors of [144] proposed ontology-services to facilitate inter-operability among agents. By defining this ontology they provide a vocabulary to be used in the communication among different agents. We found a similar approach to ours in [226], the authors present an overview of issues concerning allocation of processes in the grid through negotiations. An ontology whose aim is to share information between sending and receiving agents in Multi-Agent System is presented in [6]. The challenge targeted by the authors was the interactions among agents as agent-agent and agent-user communication can be very complex. Van Aart et.al.[212] constructed a theoretical framework for message-based communication between agents. As in the current paper, the meaning and intention of the messages is specified through a message content ontology. Also, in paper [95] the author discusses the implementation of information flows and data transformations.

1.1.2 Multi Agent Systems

In order to achieve a higher automation level, especially in operation automation, Multi Agent Systems have been developed and employed. The versatility of software agents presents a serious advantage, as they can be easily developed and/or adapted to assist both buyers and suppliers in the search and selection of goods.

Our choice of using a MAS is also based on its the prior successful use in many real life scenarios and applications like: ambient intelligence [92], managing roadway network congestion [2], situation assessment problems found in environmental management applications [11], etc. The commonality of the above scenarios stands in the fact that no one entity can hope to achieve all the goals or just one goal of such complex systems; hence, several entities (agents) need to coordinate and share information to achieve the goals. This is also the case of freight transport, as there is no one transport provider that can satisfy all transport requests.

According to [29] agents are fundamentally a form of distributed code process and thus comply with the classic notion of a distributed computing model comprising two parts: components and connectors. Components are consumers, producers and mediators of communication messages exchanged via connectors. FIPA-ACT theory states that messages represent actions or communicative acts

(i.e. speech acts or performative). The agents must be designed to allow fluent and flexible natural language communication in a manner similar to human negotiators. A communication language between agents is used as an underlying mechanism that ensures coordination and interaction of agents.

An Agent Communication Language (ACL) is a language with a precisely defined syntax, semantics and pragmatics that is the basis of communication between independently designed and developed agents. ACL standard allows encoding/decoding of information exchange by the agents. The most popular ACLs are: Foundation for Intelligent Physical Agents-Agent Communication Language (FIPA-ACL) [181] and Knowledge Query and Manipulation Language (KQML) [81]. These languages/standards are based on the theory of *Speech Acts* [197].

Another advantage of using a MAS in our framework is that it can provide the perfect setting for developing negotiation models. An agent can negotiate with other agents existing in the same environment. Based on information exchange between agents, by using messages, the negotiation process can take place. In order to reach agreements, agents need to speak a common language and to follow a common set of rules.

As far as our knowledge and literature research goes, there are few MAS-based approaches in the brokering of freight transportation. A noteworthy approach, similar with our system, was proposed by M.R.K. Mes et.al. in [151]; the authors used MAS to develop a real-time solution for scheduling time-sensitive full truckload pickup-and-delivery.

1.1.3 Negotiation

Negotiation Generalities

Negotiation is a complex dynamic process by which two or more parties recognise the differences of interest and values existing among them and seek a compromise agreement. According with [135] the core of negotiation is reciprocal offer and counter-offer, argument and counter-argument in an attempt to agree upon outcomes mutually perceived as beneficial. Generally, negotiation brings together three topics [120]: *negotiation subject*, *negotiation protocols*, and *negotiation strategies*. In our case the *negotiation subject* is represented by a freight transport request,

the other two will be detailed further along this section.

Automated negotiation is possible through the use of software agents acting in a dynamic and unforeseeable environment. The negotiation process implies the following steps (not necessarily in this order) between the actors: establish communication, receive/send offers (i.e. proposals), evaluate offers, make concessions, impose conditions, find trade offs, accept or reject offers, formulate counter-offers, close the deal or give up.

Usually, negotiation is an open process between two parties aiming to find an acceptable solution to a non-trivial conflict. In order for the negotiation process to achieve the best results, some of the following situations need to be reached:

- conflicts consist of two or more parties/groups.
- at least a major conflict exists.
- all parties feel that the negotiation process will lead to a better outcome.
- all parties want to work together.

We consider that all the above situations are reached in the proposed freight brokering scenario. Three parties are involved in our scenario: the cargo owners, the transport providers and the broker. At least one conflict exists, as various transport providers have conflicting offers. As there is no other way to establish transport contracts among parties, the negotiation process is the only manner through which all parties can gain something and therefor must work together. Hence, the use of negotiation in our system is motivated and necessary. Also, automated negotiation plays an important role in dynamic transactions in today's commerce, see [49, 116].

According with [120] there are three main approaches to negotiation: game-theoretic, heuristic and argumentation-based. The game theoretic approach uses formal mathematical models to describe, recommend or predict the actions parties take in order to maximise their own profit. However, reaching an equilibrium by using optimal strategies may be computationally hard to achieve, thus representing a significant disadvantage. *Heuristic* approaches are used to overcome the disadvantages of game theoretical approaches by looking for heuristic solutions.

Although, these heuristic solutions are not always optimal and are often tested through experiments [193]. In *argumentation-based* approaches agents can offer some arguments about their decisions [135]. Thus other actors can prepare informed counter-proposals, leading to a faster negotiation process and fewer failed negotiations. An overview of the state of the art in negotiations with a focus on complex negotiations and their challenges is presented in [193].

Negotiation Protocols

The interaction protocols are a set of rules with the aim of establishing communication between participants. Negotiation protocols are a subset of interaction protocols that govern the negotiation process. Among the advantages (for the *freight broker*) of using automated negotiation protocols versus classical negotiation (i.e. human based) we mention: lower waiting times, evidence based decisions, easy adaptability to various scenarios and cost optimisation (i.e. financial, resources).

Contract net (CNET) represents a task allocation protocol that is standardised by the FIPA[1]. It was initially developed by Smith [204] for allocating tasks in a distributed environment. The algorithm specifies two roles that agents can take: *manager, contractor*. A *manager* can allocate tasks to a *contractor* in case he is not able to perform those tasks. The allocation mechanism is based on bids (offered by the contractors). The *contractor* with the best bid will receive the task, based on the requirements established by the manager.

Iterated Contract Net (ICNET) represents an improvement of CNET protocol, also standardised by the FIPA[2]. The major contribution of ICNET is that it allows multiple rounds of bidding before reaching an agreement between the parts. The stop condition for the algorithm is met when either all offers were rejected or either one offer was accepted by the broker.

A special approach to negotiation is the *task allocation* problem. In this approach, agents are able to perform tasks with associated cost and sometimes delegate tasks to other agents. The delegation of tasks is due either to the impossibility of performing (by the current agent) or due to a lower cost.

[1] http://www.fipa.org/specs/fipa00029/SC00029H.pdf
[2] http://www.fipa.org/specs/fipa00030/SC00030H.pdf

Another protocol derived from CNET is *Dynamic Contract net* (DynCNET) [228], which is able to dynamically assign tasks. Like CNET, upon reaching an agreement, an agent is bound to respect it. However, if a new opportunity occurs before the task is started, i.e. the agent receives a better offer, the agent can allocate the task to another agent through a new biding process.

Among other protocols used in the negotiation process we mention: *Rubinstein's alternating offers protocol* [171] and the *Monotonic Concession Protocol* [74][42].

In [44] the authors studied the effect/impact of dynamically changing the negotiation mechanism over the utility function, in a disaster management information system.

1.1.4 Complex Networks Analysis

The establishment of transport contracts between transport providers and cargo owners enables them to interconnect into a large social network of transport stakeholders. As new transport requests are processed by the broker, a highly interconnected network emerges. Hence, we can use some of the information related to this inter-connectivity of agents to provide additional leverage for the broker during the negotiation processes. The inter-connectivity emerging between pairs of agents is called *social behaviour*, since the behaviour on one agent can influence the other agent.

To the best of our knowledge, we are not aware of any previous studies that augment an automated negotiation protocol with information extracted from the resulting interconnected world of the actors involved. Nevertheless, we could find some related studies, as follows. Van Doosselaere was able to infer the social rules that brought the rise of capitalism by analysing the link between commercial agreements and social processes ([222]). Further more, Money R. B. has shown that social activity influences human based multilateral commercial negotiations ([159]). Nan S. A. highlighted the potential impact of using social structures (networks) in conflict resolution processes through negotiation ([160]). According to [56], CNA can be used to support strategic collaborations. Greve A. et al. have used CNA to discover that social capital is the most important factor in productivity ([99]). The authors of [73] have made a marathon in their textbook to support the utility

of CNA in various areas including: game theory, auctions, bargaining, etc. Thus, we argue there is sufficient evidence to motivate a research study into social automated negotiation.

1.1.5 Previous Work

This chapter complements some preliminary results reported in [138], where the authors proposed an initial system architecture for a multi-agent system that provides a brokerage service with intelligent logistics. The goal of the system is to improve the competitiveness of logistics companies by increasing the quality of provided logistics services and by reducing logistics costs.

The context of our work we has two types of customers, on one hand cargo owners (named *Cargo owners*) and on the other hand providers of transport (*Transport providers*). These customers are represented in the system by agents such as *aCAgent*, respectively *aFTPAgent*. Other agents are contributing to the functioning of the system are *aFBAgent* (Freight Broker Agent) that represents the transport brokering service and *aFBRAgent* (Freight Broker Registry Agent) that manages the requests of customers and transport providers.

In the follow-up paper [137] an approach for semantic representation of relevant information to freight transport exchanges was introduced. Thus, facilitating the connection of cargo owners with freight transportation providers through the broker in order to conclude a contract. The broker has a unique intermediate position between buyer and seller of transportation services. It is a self-interested agent with the goal to sell freight transport services at the highest possible value (*oval*) to cargo owners and to buy freight transport services from transport providers at the lowest possible value (*tval*). The business model of the broker implies a positive difference between $oval - tval$ in order to support broker's operating costs and produce its profit.

The system diagram illustrating the types of agents and their acquaintance relations are presented in Figure. 1.1.1. In [138] you can find more details about description and comportment of all agents mentioned above. The diagram is of UML (Unified Modelling Language) type, while agent types are represented using *agent* stereotype. Agent acquaintance relations are represented as UML asso-

ciations [86].

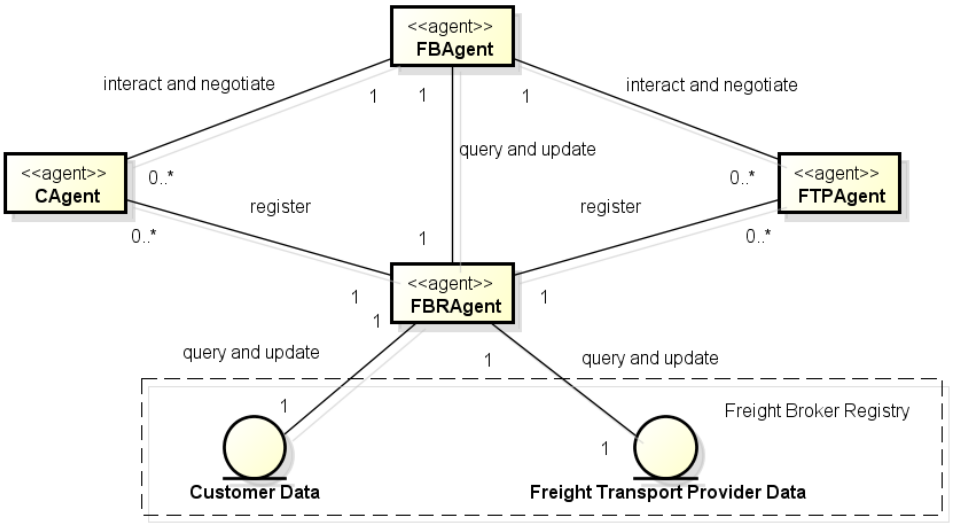


Figure 1.1.1: MAFTLS System Diagram

1.2 Semantic Modelling of Information

1.2.1 Updating the Semantic Model

As mentioned before in a previous work [138] the semantic modelling of information related to MAFTLS was done through the use of four ontologies, using as skeleton the *InterLogGrid* [195] ontologies. The four proposed ontologies are: Transport Request Ontology, Transport Resource Ontology, Freight Ontology and Messages Ontology. Transport Request Ontology describes the elements of a transport request issued by a logistic customer, Transport Resource Ontology describes transport resource capabilities provided by logistic companies, Freight Ontology describes the freight that must be transported and Messages Ontology is used for describing the messages exchanged by agents in the system. Therefore, the main concepts of the logistics ontology are Logistics Customer, Logistics Company, Transport Service, Logistics Resource and Logistics Object.

The aim of this ontologies was to provide the necessary vocabulary for the development of a freight transportation broker with the purpose of connecting cargo

owners with the freight transportation providers through appropriate contracts.

In paper [139] the ontologies have been extended and updated for a better representation of MAFTLS. Therefore new classes were introduced, transformed and reused existing ontologies such as: *AddressHomeLocation*[3], *WeightOfFreightUnitOfMensurement*[4] or *MarketPriceOfFreightCurrency*[5]. Yet another issue that we tackled in this update was determining which of object property or datatype property were more appropriate for describing the properties of MAFTLS concepts.

Some examples of updates are mentioned as following. In the *Transport Request Ontology*, the classes *DateOfCharge* and *DateOfDischarge* become data type properties i.e.: *hasStartMomentOfCharge* and *hasFinishMomentOfDischarge*. Moreover, the class *Product* was renamed into the class *Freight* from the *Freight Ontology*. Also, in *Freight Ontology* the classes *TimePeriod*, *ExpirationDate*, *DescriptionContent*, *NameOfFreight*, *LogisticsLocations* (including *PointOfDestination* and *PointOfDispatch*) and *QuantitativeValue* were removed. Instead, we added new properties (e.g. *hasDescriptionOfFreight* and *hasTypeOfAnimal*).

For the development of the ontology we used Protégé[6] – one the most popular ontology development environments and for presentation of ontology we prefer to use the Manchester OWL syntax [112], which is more readable and user friendly for non-logicians, as well as supported by ontology editors such as Protégé. You can see an example of a transport request in Listing 1.1 using Owl Manchester representation.

1.2.2 Usage Scenarios

Now, let us present a usage scenario of transport resource descriptions that use the proposed ontologies. We start with a basic simple case and then expand it into a more complex scenario with more transportation options.

Let us assume that there is a request for transporting flat doors from the city of Sibiu to Bucharest issued by *Trans LTD*. Such a request is contained by a message exchanged by the Customer Agent – *CAgent* and the Freight Broker Agent –

[3] http://daml.umbc.edu/ontologies/ittalks/address
[4] http://idi.fundacionctic.org/muo/muo-vocab.html
[5] http://www.daml.ecs.soton.ac.uk/ont/currency.daml
[6] http://protege.stanford.edu

FBAgent in MAFTLS. The message is expressed using the underlying Agent Communication Language that is abstracted away in this section and introduced in section 1.3. We assume that the request is issued on December 30th and it is available, i.e. "pending", for 24 hours. The goods must be picked up on January 1st and have to be discharged no later than January 2nd. In order to transport the flat doors we are constrained to use pallets. The total weight of the transport is 3000kg. No special freight transport conditions are necessary, as the doors are made out of solid materials that are not perishable.

The vehicle chosen for this job has a width of 2.55 meters, a length of 12 meters and a height of 4 meters. The total transport volume that this type of vehicle is able to carry is of 60 cubic meters. The Diesel truck used is parked nearby the city of Sibiu, in Şelimbăr village.

For this base case several issues are ignored. For example, we do not consider the fact that the truck has to return to its initial home location (Şelimbăr, 3 Principal St.) Also we do not consider any legal constraints that may delay the transport, e.g. a truck driver is not allowed to drive for periods longer than 4 hours without a break. You can see a representation of the request in this scenario in Listing. 1.1.

Note that in more realistic scenarios, the vehicle used for transportation has to return to its home location before a reasonable deadline. This time interval depends on the existing contract agreed by the brokerage company and the freight transportation provider. In order to reduce transportation costs, the brokerage company has to ensure that the movement of the vehicle without cargo is kept minimal. In our case the broker has to study all possible comeback routes from B (Bucharest) to SB (Sibiu). If there are no suitable comeback routes then the vehicle must be sent back to SB without cargo. The main problem is finding matching cargo for the vehicle.

```
1  Individual: lob:#request1
2  Types:
3          lob:#LocationPoints,
4          lob:#PointOfDestination,
5          lob:#GpsCoordinatesDestination,
6          lob:#AddressDestination,
7          lob:#PointOfDispatch,
8          lob:#GpsCoordinatesDispatch,
9        lob:#AddressDispatch,
10       lob:#Request,
```

```
Facts:
    lob:#hasPointOfDestinationName    "PRACTIC LTD",
    lob:#hasStartMomentOfCharge    "02.01.2015",
    lob:#hasOwnerID    "1",
    lob:#hasRequestID    "1",
    lob:#hasDestinationLongitude    "26.007344",
    lob:#hasDispatchLongitude    "24.178083",
    lob:#hasStartMomentOfCharge    "01.01.2015",
    lob:#hasDispatchLatitude    "45.773270",
    lob:#hasFinishMomentOfCharge    "01.01.2015",
    lob:#hasFinishMomentOfDischarge    "02.01.2015",
    lob:#hasDestinationLatitude    "44.410910",
    lob:#hasTTL    "03.01.2015",
    lob:#hasOwnerName    "TRANS LTD",
    lob:#hasPointOfDispatchName    "TRANS LTD",
    lob:#hasDateOfRequest    "01.01.2015",
    lob:#isJuridicPerson    "True"
```

Listing 1.1: Example of a transport request in Owl Manchester representation

Let us expand the base case with diverse constraints. Table 1.2.1 presents detailed information about the transportation requests available for a certain time period. We assume that the requests and the time allocated to them is in accordance with Romanian transportation laws. We take into consideration 10 Romanian cities and thus we created a 10 by 10 matrix. Each matrix element can contain data about the day of departure and arrival of a potential trip, but only a part of these elements are usually filled-in.

For example, let us consider the element at the intersection of row SB with column B. It should be interpreted as follows: there is a freight request departing from town SB on day 01 that must arrive in town B on day 02. To simplify reading we shall use the words *vehicle* and *truck* interchangeably as an abbreviation for *logistic transport vehicle*. Furthermore, we suppose that the truck's home location is not in the city of SB, but in a nearby city called SE (Şelimbăr). The truck can travel between SB and SE in less than an hour. Initially the vehicle is dispatched from SE and sent to SB, as SE it is the closest vehicle base near SB.

Before continuing with the description of the scenario, we must clarify the other abbreviations denoting other cities as transit points used in this scenario: CR – Craiova, TM – Timişoara, BV – Braşov, IS – Iaşi, CJ – Cluj, CT – Constanța and BZ – Buzău. All these acronyms represent cities, excepting BZ which repre-

Table 1.2.1: Enhanced transport usage scenario

	SB	SE	B	CR	TM	BV	IS	CJ	CT	BZ
SB			01-02							
SE										
B				02-03		03-04	02-04		02-04	
CR					03-04					
TM	04-05							03-04		
BV	04-05									
IS	04-06									
CJ		04-05								
CT										04-05
BZ	05-07									

- This option represents the optimal choice, as there are no stationary days or long itinerary without any cargo;
- This route implies a stationary day in B (day 02);
- This colour represents a transport option that is not valid, as the cargo type from IS to SB can not be transported with the current vehicle;
- Although this route returns back to SE, it implies an itinerary from B to CJ without any cargo;
- This option is valid and returns the truck in SB, but the time needed (7 days) exceeds the available time for the truck (6 days.)

sents a farm in the county of Buzău.

Let us continue on the basis of the simple scenario. Once arrived in town B, see Table 1.1, there is no direct return possibility to SE or SB. A possible return trip involves sending the vehicle from B to CT on days 02-04 (see the blue cells in

Table 1.1). Furthermore, a cargo request from CT to BZ (day 04-05) and another from BZ to SB (day 05-07) could also be serviced. However, although the truck would return to SB, which is nearby the vehicle home location, this route is not eligible. We assumed that the truck is only available for 6 days out of 7 days (necessary for this route). Such a constraint can be expressed in our ontology through the *hasVehicleAvailabilityStartTime* and *hasVehicleAvailabilityFinishTime* data properties, having as domain the *LogisticsVehicle* class. The aforementioned data properties define a time period when the vehicle is available for the current broker.

Another option is to send the vehicle from B to CJ such that it can take a cargo from CJ to SE (day 04-05), see the red cells in Table 1.1. Although the vehicle returns directly to its home location, the truck has to travel form B (day 02) to CJ (day 04) without any cargo. Thus the broker has to support all the expenses during this period. This option is even less profitable than the first one. Sending the truck from B to IS (day 02-04) and then from IS (day 04-06) to SB is not a valid option (see the orange cells in Table 1.1) because the freight transportation requirements from IS to SB can not be accommodated on the current vehicle. Between B and IS the medications have to be transported, which also fit onto pallets and do not require any additional conditions than flat doors. But between IS and SB frozen vegetables have to be transported, inside a frozen chamber, which our vehicle is not suited for.

Transporting a load from B to BV (day 03-04) and then from BV to SB (day 04-05) is a better option, as there is no significant movement of the vehicle without a payload. But the broker has to pay for the stationary (day 03) and for the load free trip from SB to SE, see the yellow cells in Table 1.1. This option is more convenient than the options that included a longer transport route of the vehicle without a payload, as the fuel consumption is small (SB and SE are nearby).

A better route goes from B to CR (day 02-03) then from CR to TM (day 03-04) and finally from TM to SB (day 04-05), see the green cells in the Table 1.1. Although it is a long route, it is the only option that does not force the broker to pay for stationary days or for moving the vehicle without a payload on long distances. The broker supports just the trip from SB to SE and return. Also all the other constraints referring to the type of cargo, capacity, etc are fulfilled.

In conclusion the most profitable route is: SE – SB – B – CR – TM – SB – SE.

Note that other types of goods than flat doors can be transported as long as they do not need any change to the vehicle features, e.g. parts of furniture from B to CV and home appliances from TM to SB, etc.

1.2.3 Evaluating the ontology

According to [169], an ontology can be evaluated by several criteria such as: (i) its coverage of a particular domain and the richness, complexity and granularity of that coverage; (ii) the specific use cases, scenarios, requirements, applications, and data sources it was developed to address; as well as (iii) formal properties such as consistency and completeness of the ontology based on the language in which it is modelled.

We chose an informal approach based on using specific scenarios to evaluate our ontology, as this work is a preliminary evaluation and the ontology has been designed based on proposed scenarios. Moreover, our approach can be also seen as a cross-validation step of our ontology with the initial design of the multi-agent system. The current version of our ontology is publicly available as a Github repository[7].

Competency questions were introduced by [167] to determine the scope of the ontology, thus acting as a test suite that provides value during analysis and validation stages. The resulted outcome is an overall answer to: *"Does the ontology contain enough information to answer these types of questions?"*. During the analysis stage we suggested few competency questions, for example: *"What are the main features of customer requests?"*, *"What is the transportation capacity of a vehicle?"*, *"What are the features of a request that can not be satisfied by the freight provider?"*[137]. Due to the lack of paper space we can not include all of them here, neither can we answer them directly. Nevertheless, by following the methodology proposed in the paper and the ontology it clearly results that all the competency questions were rigorously solved.

When resolving a freight request we have to focus our attention to each step of this process: i) analyse the request, ii) consider the freight features, and iii) determine the matching available resource(s).

[7] http://intelligentdistributedsystems.github.io/FreightOntology/

The choice of available resources matching with a certain transportation request is the result of a complex reasoning about the available transportation options. Moreover, a negotiation process between the freight broker represented by *aFBAgent* on the one hand and the provider of transport represented by *aFTPAgent* and/or the cargo owner (or customer) represented by *aCAgent* on the other hand, can occur. The details of this negotiation and reasoning processes are outside the scope of this section and they are the subject of sections 1.4 and 1.5.

The focus of this evaluation consists in determining if the developed ontologies can describe from simple to advanced transport requests, freight and transport resources.

The customer issues a request to the transport broker company and the request details are expressed in the ontology as follows. A transportation request is issued by filling in an input form (details are omitted here). Input data is then mapped to an instance of the *Request* class. Request properties, for example *hasStartMomentOfDischarge* and *hasFinishMomentOfDischarge*, are used to define reference time points that define the deadline of the contract that specifies when the goods must be delivered. In order to capture other details of the request we also use the following properties: *hasTTL* to represent the lifetime of the request (e.g. 24 hours), *hasStartMomentOfCharge* to represent the date of charge.

Usually tax values depend on the type of entity that request a transfer of cargo, therefore we introduced in the *Transport Request Ontology* the *isJuridicPerson* data property in order to capture also this type of information.

The location of the cargo pick up and the location of delivery are expressed in the *Transport Request Ontology*: *PointOfDestination* and *PointOfDispatch* classes. A location can be specified by giving its Global Positioning System (GPS) coordinates, using data properties as follows: *hasDispatchLongitude* and *hasDispatchLatitude*. Latitude and longitude are needed in situations when a postal address can not be determined, e.g. the dispatch is in the middle of a corn field or at the ranch. Sometimes an address identifies multiple entities creating confusion, e.g. an industrial park may have one postal address but 10 companies are present at that location. Thus we introduced two data properties: *hasPointOfDispatchName* (domain class *PointOfDispatchName*), *hasPointOfDestinationName* (domain class *PointOfDestinationName*) that represent unique identifiers (e.g. company name)

for the pick up entities and discharge entities.

The *LogisticsObject* class is used to define the features of the freight to be transported using subclasses and properties that represent freight characteristics, for example: *BoxOfFreight, hasDescriptionOfFreight, PerishabilityOfFreight, QuantifiableTraitsOfFreight, WeightOfFreightValue*, etc. The *BoxOfFreight* class is used to capture the type of goods packaging. For example, flat doors are transported on pallets, therefore the *Pallet* subclass of *BoxOfFreight* is used. For the number of packages we included the data property *hasNumberOfBoxOfFreight*. Through the data property *hasDescriptionOfFreight*, a textual description of the fright can be introduced. This can be useful, as for example in Roumania the truck driver is required to know what he or she is transporting.

If the cargo must be transported under special conditions then we can capture these special conditions using the *SpecialConditionsOfFreight* class. The following special conditions are represented in our ontology: refrigerated, oversized, dangerous, fragile and controlled temperature. Using the *ControlledTemperature* class and the *hasControlledTemperatureValue* data property we can specify the temperature required for the cargo.

The class *AggregationStateOfFreight* and its subclasses capture the aggregation state of freight, e.g. liquid, gas or solid. For the case involving transportation of flat doors we used the subclass *SolidState*. The *PerishabilityOfFreight* class has two subclasses *NonPerishableFreight* and *PerishableFreight*, thus it is easy to represent goods that have an expiration date. As flat doors have no expiration date we will use the class *NonPerishableFreight* to describe our request. Using the *WeightOfFreight* class we can capture the total weight of the cargo that needs to be transported. The *WeightOfFreightUnitOfMensurement* subclass of *WeightOfFreight* represents an ontology for weight measurements, thus different weighting systems can be used. The numeric value of the weight (double numeric type) is expressed by the data property *hasWeightOfFreightValue*.

The *LogisticsResource* class is a core concept of the ontology that describes transportation resource capabilities. The *LogisticsVehicle* class is used to describe the vehicle capabilities, as required for example by the orange route in Table 1.1. Advanced types of vehicle are supported: a bio-gas powered vehicle that can transport hazardous liquid materials in a tanker under a controlled temperature or an

oversized logs truck that has a hydraulic crane as a self charging mechanism. Dedicated vehicles are also supported by the following subclasses of class *DedicatedVehicle*: *AnimalsVehicle, CarsVehicle, GrainTransportVehicle, LogsVehicle, TankerVehicle, TruckMixerVehicle* and *ValuableTransportVehicle*.

For some locations, the accessibility of the truck may depend on its dimensions. Thus we must capture its dimensions using data properties, for example: *hasVehicleHeight, hasVehicleLength, hasVehicleWidth*. These properties have as domain the class *VehicleDimensions*. The *TypeOfFuel* class is used to represent the type of fuel used by the vehicle.

The class *VehicleHomeLocation* in *Logistics Object Ontology*, represents the parking address of the vehicle. This class has subclasses, for example *AddressHomeLocation* (an already developed ontology mentioned above) and *GpsHomeLocation* that can accurately describe the vehicle home location.

As of the above, we consider that the developed ontologies can both describe simple and complex freight transport scenarios.

1.3 Communication and Interaction

The main purpose of this section is to define and present the means of communication between MAFTLS's agents. Also, the information flow presented in [138] shall be rethough and improved here. The disscusion will be focused on two scenarios: the information flow starting with the arrival of a transport request for a cargo and the information flow upon picking up a request for adding a transport vehicle to the system.

1.3.1 Agent Communication Language

The Foundation for Intelligent Physical Agents (FIPA)[8] is an IEEE Computer Society[9] standards organisation that promotes agent-based technology. The use of their standards maximises the inter-operability across agent-based applications, services and equipment as it has been implemented in many projects. Hence, we chose FIPA-ACL as agent communication language (ACL), due to its high degree

[8] http://www.fipa.org
[9] https://www.ieee.org//

of acceptance in the agent programming community and moreover in open systems.

A basic FIPA-ACL message is composed of several parameters including a *performative* (i.e. type of the communication act of the message), the sender, the receiver and the reply-to (i.e. participants in communication), the content (i.e. the content of the message), the used ontology (i.e. description of content) and others. Among the parameters listed above only the *performative* is mandatory. A conversation between agents is defined as a sequence of messages exchanged by them. Control of conversation is done through a protocol parameter that defines the interaction protocol in which the ACL message is generated. Certainly one of the most complex tasks in defining the conversation between agents, is to study all the possible sequences of messages exchanges that can be performed during the conversation. A FIPA-ACL message example is illustrated in Listing. 1.2. More information about the structure and the language content of FIPA-ACL messages can be found in [82] and [85].

```
1 [<performative> sender: <sender>
2 receiver: <receiver> content: <content>..]
```

Listing 1.2: FIPA-ACL message structure

ACLs rely on speech act theory which defines a set of performatives called Communicative Acts. In [84] you can find more details about FIPA Communicative Act Library (CAL). To define the semantics of the FIPA ACL we used a formal language, i.e. Semantic Language (SL), called FIPA-SL [83].

1.3.2 Usage Scenarios

Following the analysis made in [138] we identified two main scenarios. The first scenario refers the possibility of allowing cargo owners to make transport requests. The second scenario should allow transport providers to add their resource (i.e. vehicle) to a freight broker.

The two proposed usage scenarios are very similar with those in [70]. In the first case a set of requirements (i.e constrains) is matched against a description of a resource, while in the second case a description of the resource is matched against a set of requirements. Let us assume that both the cargo owners and the

transport providers are already registered and logged in our system, i.e. Freight Broker's Website (FBW).

First scenario, one of the cargo owners formulates a request on the FBW to solicit transport of *flat doors* between addresses *A* and *B*. Also the cargo owner mentions in the request form that the weight of the flat doors is *3000 kg*.

Second scenario, a transport provider wants to make available its transport vehicle on the FBW. The transport resource must be a vehicle with some of the following capabilities: type of vehicle (regarding the number of axes), vehicle dimensions, dedicated vehicle, etc.

1.3.3 Agent Collaboration

In [211] agent collaboration was defined as the exchange of data among information processing agents, regardless of whether the exchange is productive or not.

Collaboration among different agents operating in the system implies communication between them, which is a fundamental characteristic of MAS. During research we discovered challenging problems that arise when agents try to communicate and collaborate with each other in order to reach and agreement (i.e contract). In our case an example of such collaborations is the price negotiation, transport time and quality. According to [219] the following question emerge in the design phase of such systems:

- How should messages be generated, transmitted and represented?
- How can the content of messages be standardised?
- What principles (e.g. concepts, mechanisms and patterns) can be used?
- What heuristics and guidelines tell us when to apply what principle?

Considering the above mentioned, communication diagrams were developed for the two usage scenarios, presented in Figures. 1.3.1 and 1.3.2. During the design we identified some similar features of ACL messages such as intentions (e.g. request, agree, forward, inform, query, search, response), attendees (*aCAgent, aFBAgent, aFTPAgent* and *aFBRAgent*), a content (i.e. the certain information that is

exchanged), content description and conversation control (e.g. communication protocol).

Let us now focus our attention on the flow of information that is necessary to facilitate the two proposed scenarios mentioned in section 1.3.2.

1.3.4 Information Flow

In MAFTLS information exchange takes place separately between agents that represent the *Freight Broker* (aFBAgent) and each agent that represent the *cargo owners* (aCAgent), the *Freight Broker* (aFBAgent), the *Freight Transportation Provider* (aFTPAgent) and the *Freight Broker Registry* (aFBRAgent).

The two main interaction scenarios are: (i) the cargo owners, represented by their *aCAgent*s make a transport request and pay for provided service through the broker and (ii) the transport providers, represented by the *aFTPAgent*s want to earn capital by lending their resources to the broker.

We assume that in both scenarios the customers, the cargo owners and the transport providers are logged into the system as in [117]. The interaction between an agent in the system and an external agent is not the aim of our current paper, this subject is presented in [94].

Figure. 1.3.1 represents the process taking place upon the arrival of a request from a cargo owner on the Freight Broker's Website.

Listing. 1.3 presents in an algorithmic detailed fashion the above mentioned figure. The first line of the algorithm represents communication (i.e. interaction) between a human user and its respective agent in our system, *aCAgent*. Next the *aCAgent* forwards the request to *aFBAgent* using the *messageForwardRequest* class from our ontology. On the third line, the *aFBAgent* sends a query message to the *aFBRAgent*. The query is an interrogation of the database for retrieving all matching vehicles, based on the constraints defined by the cargo owner. If matching vehicles could be found, the list of vehicles is sent by the *aFBRAgent* to *aFBAgent* (lines 4-8). The result of the negotiation process, either success of failure, is sent to the collaborating agents (lines 9-19). Else if not matching vehicle could be found the *aFBAgent*, *aCAgent* and *CargoOwner* are notified with mismatch messages.

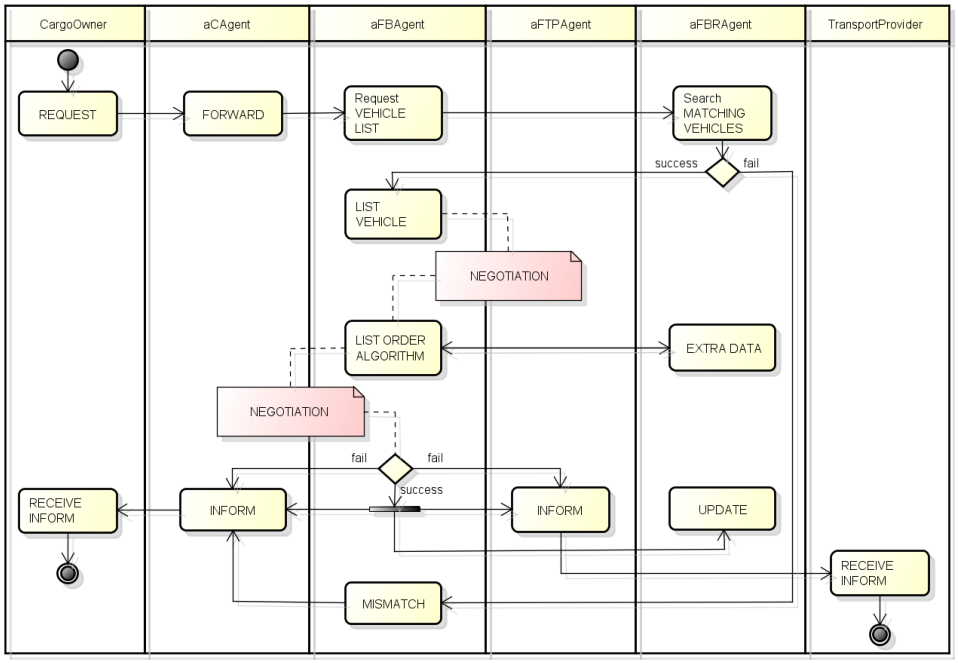


Figure 1.3.1: Communication diagram for a transport request made by the cargo owner to the broker

Figure. 1.3.2 presents a sequence of actions started by the transport provider, represented by the *aFTPAgent*, which desires to add a transport vehicle on the FBW. The detailed version of the above mentioned figure is presented in Listing. 1.4 in an algorithmic fashion.

1.3.5 Message Structure

Here we propose a message structure based on FIPA-ACL. As an example of a request communicative act, consider agent *aCAgent* (representing the cargo owner) requesting agent *aFBAgent* (that represents the Freight Broker) to process an order from "TRANS LTD" consisting of *10 pallets* with *flat doors* to be transported from *Sibiu* to *Bucharest*. This offer is only available for the next *24 hours*. The request was

```
CargoOwner->aCAgent
aCAgent->aFBAgent: messageForwardRequest
  aFBAgent->aFBRAgent: messageToDbQueryRequest
    IF success in finding matching vehicles
      aFBRAgent->aFBAgent: messageFromDbDataStructureLMV
      aFBAgent->aFBRAgent: messageToDbQueryExtraData
      aFBRAgent->aFBAgent: messageFromDbDataStructureLMVP
      aFBAgent  negotiation process with aCAgent:
      IF negotiation success:
        aFBAgent->aCAgent: messageOfResultRequestVehicleSuccess
        aCAgent->CargoOwner: RequestVehicleNotificationOfSuccess
        aFBAgent->aFTPAgent: messageOfResultRequestVehicleSuccess
        aFTPAgent->TransportProvider: RequestVehicleNotificationOfSuccess
        aFBAgent->aFBRAgent: messageToUpdateInfoVehicle
      ELSE negotiation failed:
        aFBAgent->aCAgent: messageOfResultRequestVehicleFailed
        aCAgent->CargoOwner: RequestVehicleNotificationOfFailure
        aFBAgent->aFTPAgent: messageOfResultRequestVehicleFailed
        aFTPAgent->TransportProvider: RequestVehicleNotificationOfFailure
    ELSE failure in finding match vehicles
      aFBRAgent->aFBAgent: messageOfResultRequestVehicleMismatch
      aFBAgent->aCAgent: RequestVehicleNotificationOfMismatch
      aCAgent->CargoOwner: RequestVehicleNotificationOfFailure
```

Listing 1.3: First scenario of message exchange between agents

```
TransportProvider->aFTPAgent
aFTPAgent->aFBAgent:messageForwardRequest
FBAgent->aFBRAgent:messageToDbUpdateAddVehicle
aFBRAgent ->aFBAgent: messageFromDbRegistrationResponse
IF registration success:
  aFBAgent->aFTPAgent: messageOfResultAddVehicleSuccess
  aFTPAgent->TransportProvider: AddVehicleNotificationOfSuccess
ELSE registration failure:
  aFBAgent->aFTPAgent: messageOfResultAddVehicleFailed
  aFTPAgent->TransportProvider: AddVehicleNotificationOfFailure
```

Listing 1.4: Second scenario of message exchange between agents

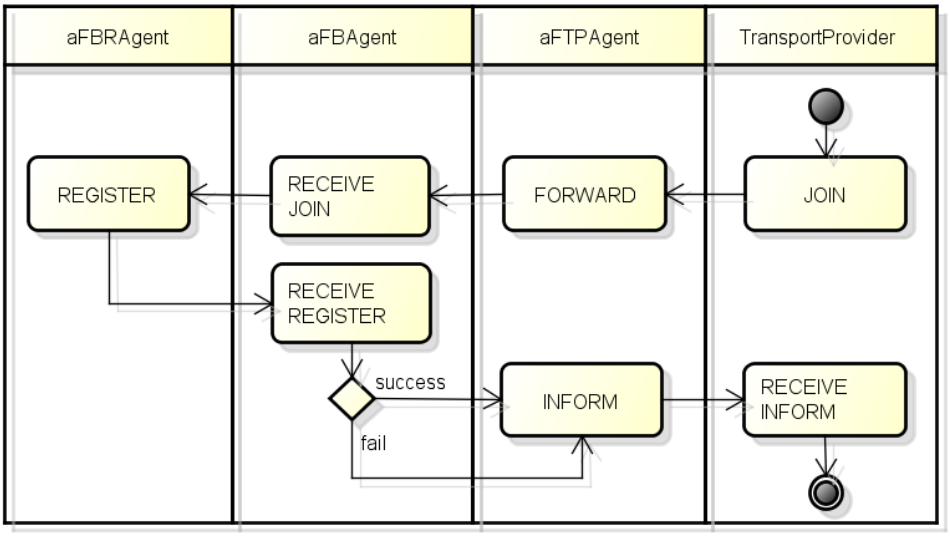


Figure 1.3.2: Sequence diagram for adding a freight provider in the broker's system

made in *2015.05.01*, date of charge was on *2015.05.03* and date of discharge was on *2015.05.04*. In FIPA notation, we express the request as in Listing. 1.5.

```
(request
  :sender aCAgent
  :receiver aFBAgent
  :reply-by: hasTTL=24:00:00
  :content (deliver hasOwnerName = "TRANS LTD"
    hasDateOfRequest="2015.05.01"
    hasNumberOfBoxOfFreight="10"
    hasNameOfFreight="flat doors"
    hasStartMomentOfCharge="2015.05.03"
    hasStartMomentOfDischarge="2015.05.04"
    PointOfDispatch="Sibiu"
    PointOfDestination="Bucharest")
  :reply-with request-01
  :language sl
  :ontology lob
  :conversation-id request123)
```

Listing 1.5: Example of a FIPA-ACL request message

```
(agree
```

```
2    :sender aFBAgent
3    :receiver aCAgent
4    :reply-to request-01
5    :language sl
6    :ontology lob.message
7    :conversation-id request124
8 )
```

Listing 1.6: Example of a FIPA-ACL agreement message

We mention that the freight is transported on pallets, a pallet can transport a maximum of 5 flat doors. To express the quantity we use the class *BoxOfFreight* that represents the wrapping of the freight. To capture the number of boxes we use the property *hasNumberOfBoxOfFreight*.

An example where an agent accepts the request is presented in Listing. 1.6.

1.3.6 Message Content Ontology

Ontologies play a significant role not only in the agent communication, but also in knowledge capturing, sharing and reuse. One of the main reasons why ontologies are being used is semantic inter-operability. Thereby, as we specified in [140] we have decided that the request, the freight, the transport resource and the messages exchanged between agents used by the system must be semantically represented.

When agents want to communicate, an appropriate message content ontology is selected. This ontology is used by agents to make conversation about issues related to specific domain.

FIPA communication stack can be separated into seven sub-layers such as: transport, encoding, messaging, ontology, content expression, communicative act and interaction protocol.

Considering that although FIPA allows for the use of ontologies when expressing messages content, it does not specify any particular representation for ontologies or provide any domain-specific ontologies. In this section we propose to provide a messages ontology for our multi-agent system. This ontology was generated considering initial architecture presented above. In fact this proposed messages ontology is a vocabulary that help us to express the content of messages exchanged between agents in the system. For the development of the ontology we followed an engineering methodology as in [167].

First, we need to define some specific terms of scientific literature used from now such as: ontology, message, content, conversation and protocol.

An *ontology* is used to represent knowledge that is shared between different entities. It provides terms and vocabulary used to represent knowledge so that both sender and receiver can understand.

According to [180] a *message* is an individual unit of communication between two or more agents. A message corresponds to a communicative act, in the sense that a message encodes the communicative act for reliable transmission between agents. Note that communicative acts can be recursively composed, so while the outermost act is directly encoded by the message, taken as a whole a given message may represent multiple individual communicative acts.

A *content of the message* is a part of a communicative act and denotes the content of the message. In [82] the meaning of the content of any ACL message is intended to be interpreted by the receiver of the message. This is particularly relevant for instance when referring to referential expressions, whose interpretation might be different for the sender and the receiver.

A *protocol* is a special set of rules, used by end points in a connection, in order to establish and maintain communication. Protocols specify interactions between the communicating entities. Further information about protocols and agents communication are defined by Amit K. Chopra's [53].

In our scenario, we use the ontology to define the types of message exchange between agents, this completes our already developed ontologies. All the ontologies developed through this research project are available online[10].

The top level class of our ontology is the *message*, see Figure. 1.3.3, which in turn is also a *Thing*. We then dived the message in other 4 subclasses: *messageForward*, *messageOfResult*, *messageToDb* and *messageFromDb*. The separation was based on the the general purpose of the messages in our system.

The *messageForward* class describes messages that are being forwarded, the message content does not change, only the sender and receiver change. Three other sub classes of *messageForward* are defined here: *messageForwardRequestVehicle* related to the first scenario, *messageForwardRequest* related to both scenarios defined in section 1.3.2, and *messageForwardAddVehicle* related to the second sce-

[10] https:\intelligentdistributedsystems.github.io/FreightOntology/

nario. E.g. *AddVehicleNotificationOfFailure* is a subclass of *messageForwardRequestAddVehicle* therefore this subclass is related to the second scenario. Further we see that this subclass is equivalent to *messageOfResultAddVehicleFailure* which is a message sent by *aFBAgent* to *aFTPAgent* when a vehicle could not be added to *aFBRAgent*, but our current subclass has as sender *aFTPAgent* and receiver *TransportProvider*. Thus *AddVehicleNotificationOfFailure* is a forward message.

The class *messageOfResult* defines messages sent by *aFBAgent* after the process of negotiation (first scenario) or after *aFBRAgent* has received the request to add a vehicle in the system data base (second scenario). Otherwise said these messages conclude the processes in both scenarios. These type of messages are then forwarded through the *messageForward* messages.

All the messages received by *aFBRAgent* are defined by the *messageToDb* class. Also this class has two other subclasses: *messageToDbQuery* which contains only queries (i.e. interrogations), and *messageToDbUpdate* comprising of messages that update the system data base. For example, after the negotiation is finished with *aCAgent* in the first scenario and a vehicle (V) is chosen *aFBAgent* sends an update message to *aFBRAgent* to set vehicle V as occupied for the time of the transport.

The class *messageFromDb* describes messages containing structures needed in the process of negotiation (first scenario). These data structures are either obtained from the system data base or from the negotiation process with *aFTPAgent*, see *messageFromDbDataStructure* subclass. For the second scenario *messageFromDb* defines messages sent by *aFBRAgent* after the process of vehicle addition was tried. Let us suppose that a vehicle was not registered because it is already present in the system data base, then *aFBRAgent* should send a message to *aFBAgent* containing an error line *Update failure, (object already present...)*.

```
1 Class:
2   <lob#messageOfResultAddVehicleFailure>
3 Annotations:
4   rdfs:comment
5   "Message format:
6   Request request_id failed"
7 EquivalentTo:
8   <lob#AddVehicleNotificationOfFailure>
9 SubClassOf:
```

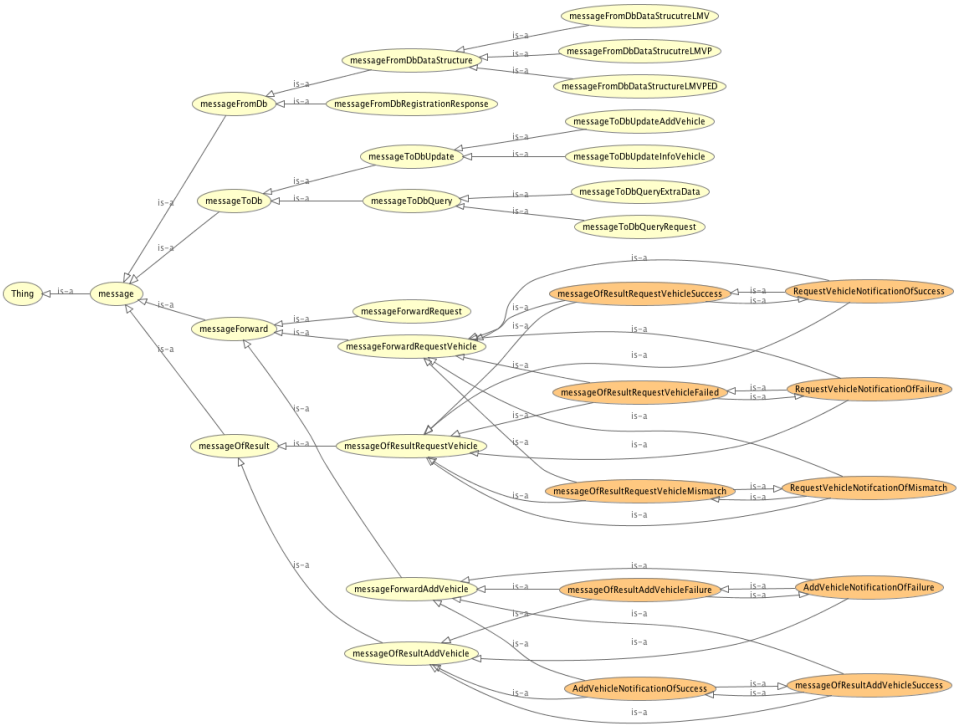


Figure 1.3.3: Infered Ontology Diagram

```
  <lob#messageOfResultAddVehicle>
DisjointWith:
  <lob#messageOfResultAddVehicleSuccess>
```

Listing 1.7: Example of a primitive class in the *Messages Exchange Ontology*

1.4 ICNET Negotiation

As of this section we have developed the system diagram, illustrating the types of agents and their acquaintance relations. Also we have established the specific objectives for each agent. Several real life use cases have been used for testing and evaluating our proposed system. Three ontologies have been developed such that we are able to define and determine the adequate transport vehicle(s) for various cargo types: *Transport Request Ontology*, *Transport Resource Ontology* and *Freight Ontology*. These ontologies have been evaluated through real life use case scenar-

ios. A 4th ontology, *Messages Ontology*, was created to define the messages exchanged in our MAS.

Brokers (in our case freight brokers) are in the unique position of being both buyers and sellers of transportation, which in many ways are in direct conflict with one another. The freight broker has two goals: to sell his service (in our case transport freight service) at the highest price possible when selling to customers (i.e. cargo owners) and to buy service (i.e. freight transport) for the lowest price possible without affecting the quality of service requested. Hence, we consider this to be the perfect setting for introducing a negotiation mechanism to resolve the above mentioned conflicts.

1.4.1 Negotiation Protocol & Mechanism

ICNET protocol

In our experiments we chose to use ICNET. The major advantage of ICNET is that the negotiation is extended to be multi-round allowing for some real bargaining [52], similar to many real life negotiations. Thus the negotiation process can be easily understood and leveraged by the involved parties (cargo owner and transport vehicle owners).

ICNET is considered to be an improvement of CNET, created as a solution to the lack of bargaining in CNET [52]. Thus from the perspective of the *freight broker* when no offer is satisfactory there is always the possibility to request a new round of offers. Thus the number of failed negotiations (no contract established) should decrease compared to CNET. Further more Scafeş et. al. have observed that ICNET provided better results [41] versus other negotiation protocols (i.e. CNET and Direct Task Assignment) when considering the perceived average utility (quality metric).

A drawback of ICNET is that it limits the number of message types that can be exchanged in relation to the CNET and therefore some interactions possible in the original Contract Net are not possible here (i.e. the freight transport provider cannot initiate a negotiation, only the broker can) [59]. However, this does not represent an issue in our case, as the *freight broker* is designed to be the only agent capable of starting negotiations

Another disadvantage of using ICNET is the higher cost of communication, estimated as the number of messages exchanged between the involved parties [43]. However, in our freight transport scenario there is no need for immediate/urgent actions, as in the case of disaster management described in the aforementioned work. Therefore a higher cost of communication does not pose significant problems in our scenario.

Negotiation Mechanism

As we have previously mentioned, there are three parties involved in the negotiation, each represented by an individual agent: the client/cargo owner (by the aCAgent), the broker (by the aFBAgent) and the freight transport providers (by the aFTPAgents). The aFBAgent acts as a double contractor in this scenario, conducting the negotiation with the aCAgent and aFTPAgents, respectively.

The negotiation mechanism is illustrated in Figure 1.4.1. The aCAgent commences the negotiation by using a discount applied to the reference price according to its personality and passing this to the aFBAgent. The aFBAgent then issues a Call for Proposal (CFP) using that value to negotiate with the aFTPAgents through the ICNET protocol. After reaching an agreement for a given value, if possible, the aFBAgent initiates a new ICNET negotiation with the aCAgent by adding its commission to the previous value. If this final negotiation is successful, the whole process is successful and the final price is reached.

1.4.2 Experiments

Experimental setup

For our experiments, we have considered a simple scenario in which there is a single aCAgent, a single aFBAgent, and one hundred aFTPAgents. With pre-filtering applied, the aFBAgent has identified aFTPAgents matching on all criteria (e.g. route, weight etc.), leaving only the price to be established.

In order to simulate the manner in which the agents conduct negotiations, we have created four possible personalities. These personalities influence the initial and/or reserve values for these agents (*High Price* or *Low Price*) and how flexi-

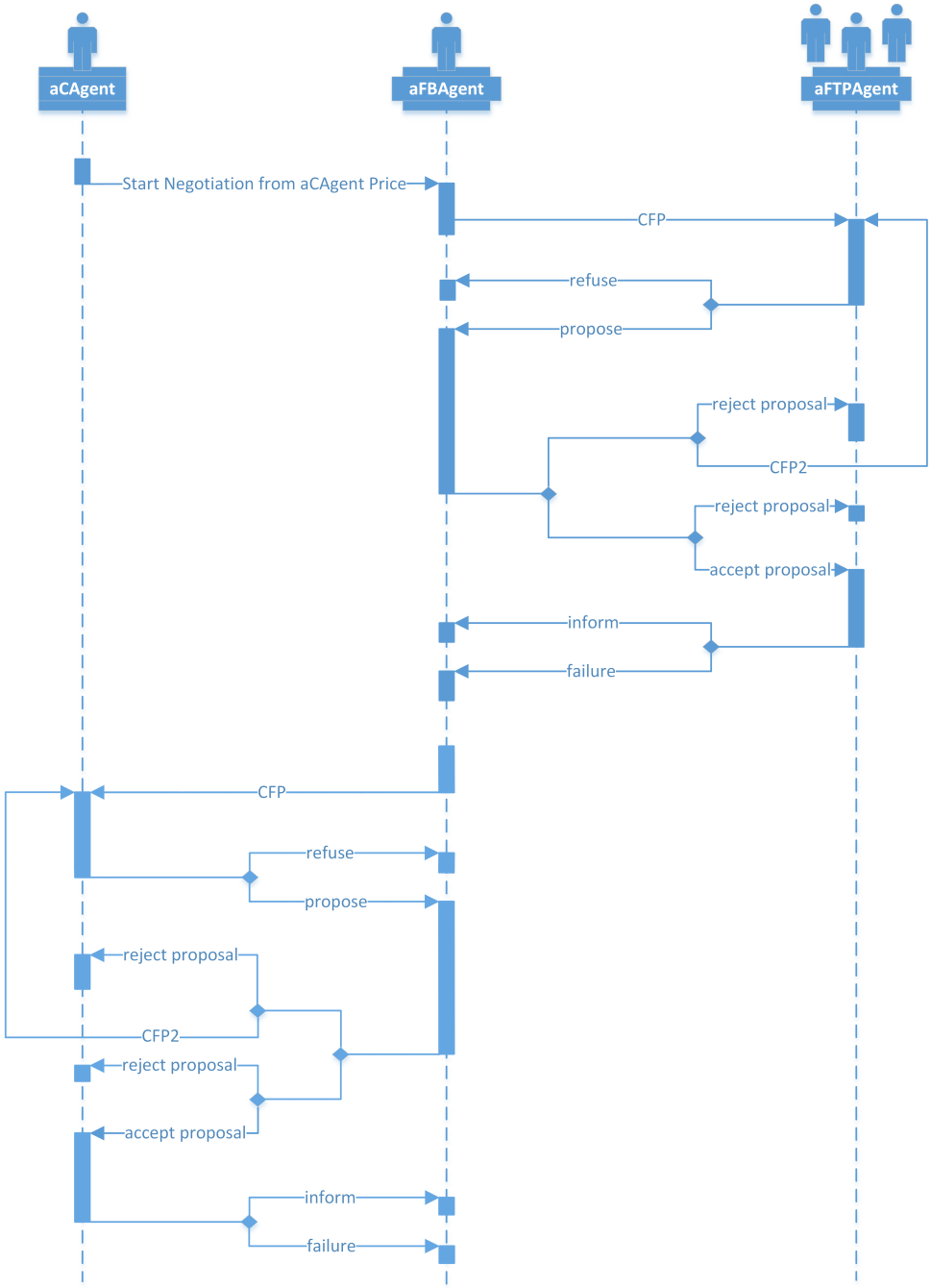


Figure 1.4.1: ICNET Negotiation Sequence Diagram

ble they are during the rounds of the negotiation (*Conservative* or *Lenient*). The values considered for the minimum/maximum initial/reserve percentages (Min/-

Max Pct) and minimum/maximum round percentage discounts (Min/Max Rnd) are outlined in Table 1.4.1:

	Min Pct	Max Pct	Min Rnd	Max Rnd
`High Price Lenient`	10%-20%	20%-40%	3%-5%	5%-15%
`High Price Conservative`	10%-20%	20%-40%	0%-3%	3%-5%
`Low Price Lenient`	5%-10%	10%-20%	3%-5%	5%-15%
`Low Price Conservative`	5%-10%	10%-20%	0%-3%	3%-5%

Table 1.4.1: MAFTLS Agents' Personalities

For our experiment, the aCAgent and aFBAgent are set to the `Low Price Lenient` personality, while the one hundred transporter agents have been divided uniformly among the four possible personalities.

The latest version of JADE[11], version 4.4, has been used for these experiments, on a machine running Java Development Kit 8.

Results & Discussions

One thousand runs of this scenario are summarised in Table 1.4.2. The total cost for the aCAgent and the aFTPAgent price are expressed in monetary units/distance units (m.u./d.u.), e.g. Euro / Kilometer. The reference price, the amount that would typically be charged, is 1 m.u./d.u. As can be seen from Table 1.4.2 the average total cost amounts to 1.04 m.u./d.u., 4% higher than the reference price. In this scenario, the average aFBAgent commission is very close to 10%, meaning that 0.94 m.u/d.u. is received by the aFTPAgents. A resolution to the negotiation is reached rather quickly, the average number of rounds being 5.70. Also, the percentage of failed negotiations (which would need to be repeated) is only 6.4%.

In Figure 1.4.2 the distribution for the winning aFTPAgent personality can be observed on the left side (a). This is dominated by the aFTPAgents with the `Low Price Lenient` trait having 47.2%, followed by `Low Price Conservative` with 41.2%. Agents of personalities `High Price Lenient` and `High Price Conservative` are only successful in 3.5% and 1.7% of runs.

[11] http://jade.tilab.com/

Result Measured	Value
Total Cost (average)	1.04 m.u./d.u.
Broker Commission (average)	10.02 %
Transporter Price (average)	0.94 m.u./d.u.
Number of Iterations (average)	5.70
Failed Negotiations	6.40 %

Table 1.4.2: ICNET Experiment Results Summary (1000 Runs)

In Figure 1.4.2 the total number of iterations needed to reach a conclusion for the negotiation is detailed on the right side (b). In this scenario, very few rounds are required (4-9), with most runs terminating after 5-6 iterations.

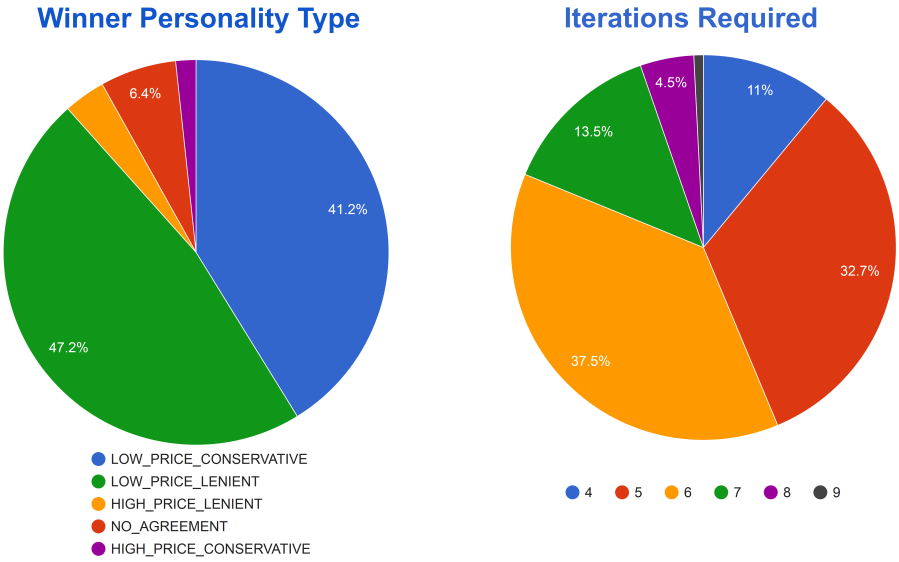


Figure 1.4.2: ICNET Results (left side) a.Chosen Transporter Personality Type. (right side) b.Number of Iterations.

Figure 1.4.3 illustrates the histogram for the total cost paid by the aCAgent. It resembles the Gaussian distribution that is expected in this scenario, with a maximum peak at 1.05 m.u./d.u., and with possible values ranging from 0.85 to 1.275 m.u./d.u.

Figure 1.4.4 depicts the aFBAgent commission histogram. The histogram is

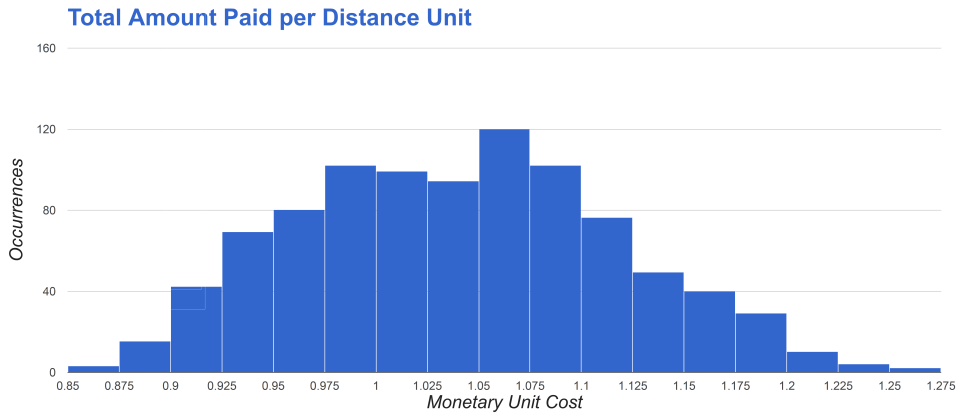


Figure 1.4.3: ICNET Total Cost (m.u./d.u.) Histogram

left-skewed due to the fact that the aFBAgent is sometimes able to achieve higher commission percentages when a better aFTPAgent price has been settled or the aCAgent is more easily satisfied.

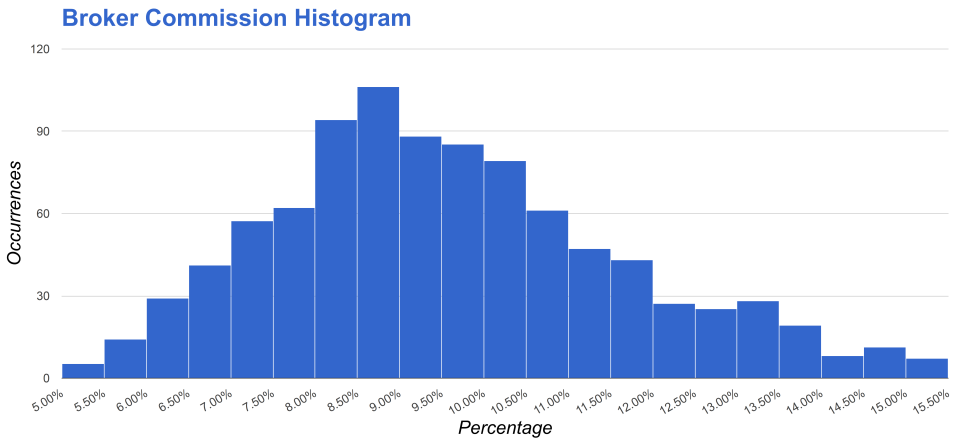


Figure 1.4.4: ICNET Broker Commission Percentage Histogram

Hence, the above experiments demonstrate that using ICNET protocol in our agent-based solution for freight transport can behave realistically and manifest similar results to human-conducted negotiation.

1.5 AICNET Negotiation

1.5.1 Introduction

In this section we propose new computational approaches to endow negotiation brokers with novel and controllable instruments aiming to better impact their business environment, by providing them with quantitative feedback regarding their decisions. Generally speaking, the vast majority of negotiation protocols only consider short-term advantages of participants, like for example monetary benefits, by aiming to increase the profit per negotiation of the broker and/or other participant agents. This is the case of our experiments with ICNET protocol, detailed in section 1.4, where only the monetary value of goods played a role in the negotiation mechanism. However, we consider that this is of limited value since the business world is far more complex and with many facets of the longer-term benefits one might gather by engaging in negotiations.

Let us consider a well known fact in economy: monopolies are bad for an open market since the broker and the entire market environment is at the hand of a single actor. The monopoly holder establishes the monetary value and the pace of evolution, see for example the case of Standard Oil [147]. Hence, brokers/markets need to prevent the rise of monopolies. This phenomenon can be controlled either through regulatory bodies and/or through active preemptive strategies, the latter approach being considered in this section. Since, monopoly hindering strategies that involve only adjustments to the monetary value are ineffective, as a monopoly will diminish the value of goods/services in order to hinder potential competition in the short term and to gain monopoly in the medium to longer term [182], a need for a broader set of actions emerges.

Hence, our motivation is to design an enhanced negotiation protocol such that the broker can dynamically and easily employ complex strategies of domain specific (transport and logistics in our case) social inspiration. For example, a strategy for hindering monopolies might consider favouring smaller / less important transporters over larger / more important transporters when their bids are the same or marginally different. Also, sometime might be better to encourage transporters that have multiple transport capabilities in order to reduce the dependence

of rather few highly specialised transporters.

Since this work heavily relies on experimental evaluation of negotiations, it was vital to create an accurate scientific representation of the negotiation environment and processes. We have successfully created a simulation system / experimental system by using concepts and computational methods borrowed from the fields of Complex Networks Analysis (CNA) [207] and Multi-Agent Systems (MAS) [79]. The simulation system was developed using Python programming language. Also, we have also utilised *Gephi* [19] for graph visualisation and exploration. The code of ES is publicly available on *Github*[12] for those interested to use and further develop it, under MIT license. We recommend using the code in Jupyter/IPython [13] interactive development environment, as it was developed to be used as such. The following subsections introduce the conceptual model of our experiments, as well as the supporting experimental system (ES).

1.5.2 Conceptual Model

MAFTLS Graph Representation

In order to analyse social characteristics of agents involved in MAFTLS, we introduce the methods used for representing the agents' social environment. CNA defines the structures used to represent such environments as networks, while SNA often refers to them as sociograms. However, both are essentially graphs (as in graph theory) augmented with specific information. Hence, we are using these terms interchangeably.

Graphs have two types of constituent elements: nodes/vertices and links/edges; nodes are the portrayal of environment actors, while links are depicting relations among the actors. In our scenario we model each *aCAgent* and *aFTPAgent* by a separate node. Links are formed among nodes that have established a transport contract. Hence, we obtain a social bipartite graph of cargo owners and transport providers, where the social relationship is based on prior commercial agreements. The broker and the broker registry are not included since they are only environment artefacts, acting as match makers, with no active role in the social relation-

[12] https://github.com/becheru/aicnet
[13] https://ipython.org/

ship formation and development. As a freight transport contract does not imply a leadership/direction, we use undirected graphs in our modelling. Also, as multiple contracts can be established between the same freight transporter and cargo owner, it is natural to augment our model with link weights.

As already mentioned, the broker is involved in two types of negotiation processes: one with the cargo owner (left negotiation in Figure 1.3.1) and the other with the freight transport providers (right negotiation in Fig. 1.3.1. Since the *left negotiation* is one to one, we consider that social factors are less important here. Nevertheless, we plan to explore the use of social information also in this case, as future work. So, we will focus only on the *right negotiation* type, where the broker is involved in one to many negotiations. Here the social characteristics of each freight transporter will be explicitly considered to differentiate between them. We are now considering possible social features for characterising freight transporters.

Measures of Social Welfare

Centrality measures are indicators of the most important/influential nodes in a social graph. The *PageRank* ([173]) is a highly utilised centrality measure in SNA studies. It stands out as the underlying algorithm of *Google*'s search engine. The algorithm for computing this measure takes into consideration the number of links of a node (similarly with *Degree* measure) and their respective *quality*. Simply put, it is important to have as many connections as possible with nodes that are also highly connected.

In our scenario, a freight transporter with high *PageRank* established many transport contracts with cargo owners that in their turn have established a significant number of contracts with other highly successful transporters. From an economic point of view, a highly rated transport provider according to *PageRank* serves the transport needs of the most active cargo owners, i.e. those cargo owners which are making most transport requests. Hence, such transport providers represent a crucial factor for the success of MAFTLS, as the revenue increases with each established transport contract.

Another measure of centrality that we are considering in our work is *Betweenness*, initially proposed by Freeman L. C. in [87] and further developed by Brandes

U. in [38]. It emphasises the nodes that act as bridges between graph communities, by representing the weak ties in a social graph, as described in [192].

Graph communities are established when a subset of nodes have a higher connectivity between themselves rather than with the rest of the graph nodes. A detailed description on how to discover such communities is given in [165]. A possible cause for the presence of communities in our scenario would imply that some cargo owners have mainly established transport contracts with the same transport providers over time, due to some specificity of their cargo that can only be satisfied by those transporters. For example, in the real world only a small percentage of freight transporters have oversized capabilities. Hence, cargo owners of mainly oversized cargo would form a community with such transporters. A transport provider with a high *Betweenness* coefficient might have a wide range of transport capabilities, as it can satisfy cargo owners that have well established communities. At least we could say that *Betweenness* can be used to highlight those transport providers that are competitive across many transport domains; the more competitive transporters, the better they are from an economic point of view.

Note that all the metrics are able to assign a quantifiable measure to each node of the social graph, hence enabling their comparison by an appropriate ranking.

AICNET Strategies

Now that we have the means to determine the social welfare of each transport provider, we must establish the goal(s) of using them. We group one or more goals into *strategies*, for a better business alignment. The broker is the agent capable of selecting and applying various strategies, according to its business and/or social interests.

It is not possible to cover all the possible strategies, even in a larger paper. Hence we will focus on few of them that we consider more relevant. We are aware that a strategy can involve more SNA metrics. However, in this work we consider only strategies that can be at least partly satisfied by a single metric. The simultaneous intertwining of multiple metrics to adhere to a specific strategy will be addressed in future works.

In subsection 1.5.1 we discussed that the existence of the monopoly might neg-

atively impact a market. Hence, being able to apply a strategy to hinder/disrupt monopolies can be of great value for the broker. One such strategy may rely on increasing competitiveness by providing an advantage to transport providers that have this trait. Let us call this strategy *Competitiveness Advantage* (*sCA*). As already mentioned, such transport providers can be highlighted using the *Betweenness* metrics. Hence, the transport provider with the highest *Betweenness* will be awarded contracts with *sCA*. The second strategy, let us call it *Page Rank* (*sPR*), will use *PageRanrk* centrality measure to identify nodes that represents potential monopolies. The freight transporter with the lowest *PageRank* coefficient will be selected as winner.

Having fixed those SNA metrics that we intend to utilise as negotiation factors, we now present some details of the winner selection in negotiation protocol (we assume the reader familiarity with ICNET – a standard task allocation protocol in MAS).

The simplest solution would be to directly (i.e. in one negotiation round) select based on the metric value. However, this approach would not imply really a negotiation. Therefore, we decided to slightly update ICNET to better support the use of our metrics. This is actually a natural development of our previous results that discussed negotiation processes in MAFTLS ([141]), as we can use those results as comparison benchmarks.

ICNET implies that the negotiation proceeds as a series of negotiation rounds until one or more freight transport providers will agree with the broker's offer or the maximum number of iterations is reached (negotiation failure). During each round the broker proposes a monetary value in exchange for the transport service that the freight transporters can: accept – thus finishing the negotiation (negotiation success), reject but continue to the next round and reject and withdraw of the current negotiation. The accept is given when the broker bid is higher than transporter bid ceiling, while the reject is given when the broker bid is lower than the transporter bid floor. In between the bid ceiling and bid floor the transport will reject the current bid, but it will proceed to the next round of negotiation. In standard ICNET, if multiple transport providers accepted the offer in the current round then the one with lowest price would win. In our variant, AICNET, the accepting transport providers would be ranked based on a SNA metric and the

winner would be the highest/lowest ranked depending on the strategy used.

1.5.3 Experimental System Architecture

We now provide some of the details of our experimental system – ES. An experiment is organised as a series of simulation rounds. Each simulation round deals with solving of multiple transport requests. Each transport request can be satisfied by a negotiation process that usually takes several negotiation iterations, with a maximum threshold established. The negotiation protocol (ICNET or AICNET), number of experiments, rounds and iterations are defined by the user at the start of the experiment and they are fixed for the whole duration of the experiment. It is noteworthy to mention that a series of simulation rounds can be continued by another as the data is not lost between rounds, unless this is explicitly requested by the user. These experiments produce and record results and information that can be obtained by querying the ES's statistical module.

The population of participant agents is automatically generated at the start of each simulation round. Currently each cargo owner is only characterised by its id, since cargo owners are not involved in the *right side* negotiation. The broker and the transport providers have a more complex structure since they are endowed with "personalities". The details on the personalities have been introduced in Table 1.4.1. The personalities influence the initial and reserve values (High Price or Low Price) and the level of flexibility of the agents during a negotiation (Conservative or Lenient). Transport providers change their personalities autonomously, between consecutive transport requests, as they reach specific thresholds or randomly. The broker personality is explicitly set by the user, while transporter personalities are set at the start of each experiment, by random selection from 4 available options.

As this experiment is focused only on the *right-negotiation*, the simulation of a transport request issued by a cargo owner is simply reduced to: i) randomly selecting a cargo owner as transport request issuer, and ii) generating a random monetary values between 1 and 10000 that represents the estimated-transport-cost. In MAFTLS, the broker-estimated-cost is computed by the broker based on the details of the transport request, to determine the broker initial bid. The broker bid

for each negotiation iteration is computed based on the previous bid value and the broker personality.

During each negotiation iteration, the transporters receive the transport request from the broker, and consequently they compute their own *transporter cost estimation*. In a realistic market we would expect that the transporter-cost-estimation of transporters is randomly distributed around the broker-estimated-cost. Hence, we compute the *transporter cost estimation* for each transport provider using Gaussian distribution, as follows. The *mean* is computed by adding the broker-estimated-cost with a *displacement* (a parameter set by the user for each experiment). It models the estimation error of the transporters, for the broker-estimated-cost, as in a real setting this value is private to the broker, so transporters do not know it exactly. The *standard deviation* is given as *displacement* over *deviation* (user defined scaling parameter).

After determining its private cost estimation, each transporter proceeds to compute the bid ceiling and bid floor, based on its personality. Bid floor and bid ceiling are updated in each negotiation iteration. Then the negotiation proceeds according to the negotiation protocol, until termination is reached. If the negotiation terminates with success then the social graph is updated by creating a new link between the cargo owner and the transport provider.

1.5.4 AICNET Experiments

Experimental Setup

ES has been developed as an object oriented program in Python which makes it very easy to use and/or adapt. The user can easily interact with ES using two classes: *Environment* and *Statistics*. The *Environment* class makes the necessary initialisation and controls the execution of the negotiation processes. The *Statistics* class is in charge of gathering data during the negotiation and presenting various statistics to the user.

ES provides the user with a wide experimentation perspective, by suitable setting of the various parameters. Because of the limited space, we focus here only on experiments with the following settings of the simulation parameters: broker personality set to Low Price Lenient, 50 transport provider agents, 1000 cargo owner

agents, a threshold of maximum 12 iterations per negotiation, displacement 10 and deviation 10.

The experiments are focused on two broker strategies (*Competitiveness Advantage* (*sCA*) and *Page Rank* (*sPR*), see Section 1.5.2), aiming to provide compelling evidence that AICNET can satisfy the goal of each strategy. Since we are restricted regarding the length of the paper, the details of using ES to run these experiments are mentioned on the GitHub page of the tool. For each parameter setting, we first run 1000 rounds of ICNET, to avoid the cold start problem for computing graph metrics, and then we run another 1000 rounds of AICNET (with ICNET in parallel), to be able to draw conclusions about the effectiveness of AICNET and compare it with ICNET. Concluding, after 2000 rounds we are able to compare the results of running ICNET (2000 rounds) with those of running 1000 rounds of ICNET followed by 1000 rounds of AICNET.

Results & Discussions

Table 1.5.1 presents some results obtained with ICNET and AICNET in the context of the broker strategies. Regarding negotiation related metrics, we can observe one major advantage of AICNET: the number of iterations per negotiations is slightly smaller, which translates in less waiting time for the cargo owner.

The gain per negotiation of the winning transporter is computed as the winning price over the initial estimate of the broker. Hence, from a business perspective there are no or minor differences between ICNET and AICNET, which we consider to be a major incentive for using AICNET. Some differences still arise, albeit not major, when we take into account the personalities of the winning transport providers. However, as stated before there is almost no impact on the negotiated monetary value. Our interpretation is that transporter providers that are more conservative and give higher initial prices have better chances to participate with success in AICNET than in ICNET, resulting in a slightly more inclusive negotiation protocol.

Regarding negotiation failures, both protocols act well. Although, as shown in [141], when the implementation is done in a distributed MAS framework (e.g. Jade), some failures may arise because of communication problems between agents

distributed on different machines. Based on our obtained results, we speculate that AICNET has a slight edge over ICNET.

Metric/Strategy	*sCA*		*sPR*	
	ICNET	AICNET	ICNET	AICNET
Negotiation related metrics				
Avg. No. of iterations per negotiation	3.186	2.084	2.268	1.633
Avg. Gain of the transporters	1.185	1.195	1.205	1.217
No. wins LOW PRICE LENIENT	1628	1544	1406	1292
No. wins LOW PRICE CONSERVATIVE	341	405	403	378
No. wins HIGH PRICE LENIENT	13	13	87	120
No. wins HIGH PRICE CONSERVATIVE	18	38	104	210
No. failed negotiations	0	0	0	0
Graph related metrics				
Avg Weighted degree	3.810	3.808	3.810	3.808
Avg path length	3.780	3.602	3.771	3.786
Diameter	6	6	6	6
No. communities	155	156	158	160

Table 1.5.1: AICNET Results of negotiation and graph related metrics

Analysing the graph related metrics for *sPR*, they might superficially suggest that they only display minor differences, and thus supporting the idea that both graphs are in fact almost identical. However, this is not true, as clearly shown in Figure 1.5.2. We can observe on that figure that the graph produced using AICNET is slightly more connected at the graph periphery, thus supporting the inclusion supposition. A major difference can be seen by looking at the nodes representing the transport providers (shown in purple). With ICNET, the diameter of the nodes varies more than with AICNET. Note that the diameters are proportional to the *PageRank* coefficient. Hence, we could argue that AICNET with *sPR* has successfully hindered the rise of monopolies. This statement is further supported by the plots in Figure 1.5.1. Both the number of wins and the *PageRank* coefficients are flatten out in AICNET. Hence *sPR* is proven to be highly effective in hindering the rise of monopolies, while having no negative impact on the monetary values and other relevant metrics.

We obtained rather similar values of graph related metrics with *sCA*. However,

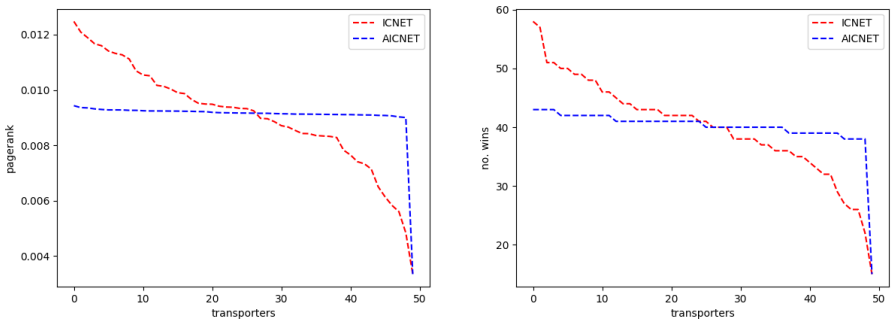


Figure 1.5.1: AICNET Plots of results obtained by applying *sPR*. (left side) *PageRank* coefficient for each transporter, (right side) the total number of wins per each transporter.

Figure 1.5.2: Plots of social graphs results obtained by applying ICNET (left side) and AICNET with *sPR* (RIGHT SIDE). Green nodes represent cargo owners while purple nodes represent transport providers. The diameter of the nodes is proportional to their *PageRank* coefficient. For plotting we used *ForceAtlas 2* algorithm included in *Gephi*

as shown in Figure 1.5.4, we can see observe different results than with *sPR*. The ICNET graph is by far more connected and there is clear evidence in the AICNET graph that at least two monopolies have risen, purple node at the top and the one at the right bottom part. Moreover, the distribution of the transport provider's wins and their associated *PageRank* coefficient are similar to power-law distributions, see Figure 1.5.3. Hence, this strategy does not hinder monopolies, rather it

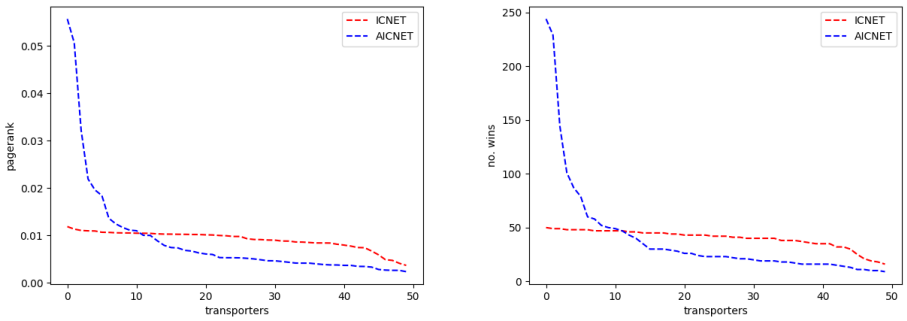


Figure 1.5.3: AICNET Plots of results obtained by applying *sCA*. (left side) *PageRank* coefficient for each transporter, (right side) the total number of wins per each transporter.

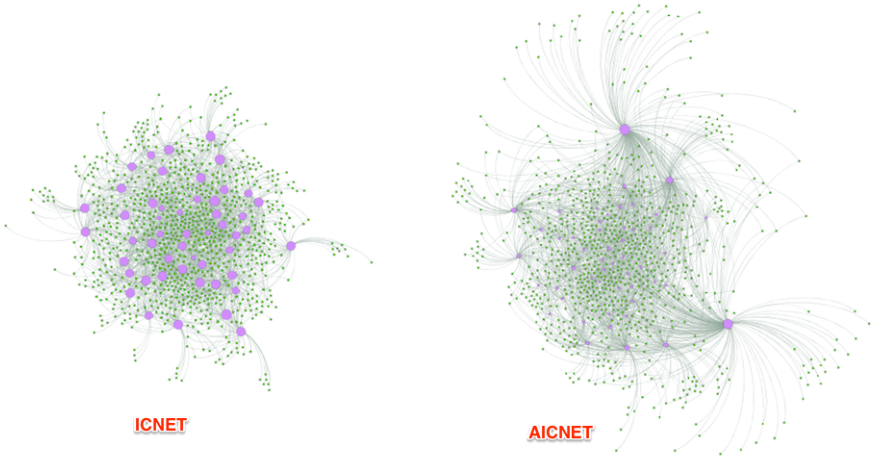


Figure 1.5.4: Plots of social graphs results obtained by applying ICNET (left side) and AICNET with *sCA* (RIGHT SIDE). Green nodes represent cargo owners while purple nodes represent transport providers. The diameter of the nodes is proportional to their respective PageRank coefficient. For plotting we used *ForceAtlas 2* algorithm included in *Gephi*.

encourages their rise.

Based on our results we can conclude that *sPR* is a good strategy for hindering monopolies, while on the contrary, *sCA* is facilitating them.

1.6 Conclusions & Future Work

Throughout this chapter we have discussed the semantic modelling of information relevant to freight transportation and proposed communication and interactions means among the agents involved. For this purposes several ontologies have been developed and tested based on MAFTLS usage scenarios. Moreover, we have established through experiments that state of the art negotiation protocols, like ICNET, can be used with success with MAFTLS in order to mitigate conflicts. Nevertheless, we introduced a novel negotiation protocol based on CNA methods, which was proven capable to implement advanced negotiation scenarios. Hence, we conclude that CNA methods and techniques have been introduce with success in the field of freight transport.

Although the present work has reached a relative maturity, there is still a lot of ground to research. One direction is especially promising, the introduction of various business strategies in AICNET.

I do not fear computers. I fear the lack of them.

Isaac Asimov

2

Tourism Portal Analysis

As during the past decades, the Internet and its functionalities have rapidly expanded and developed to become ubiquitous, so the online global and national (Romanian) tourism has developed and aligned to the current networking advancements. Presently, complex tourism services, like for example: travel, accommodation, and ticketing, can be accessed and combined using Internet. Furthermore, the Internet provides information about tourism related businesses, as well as easier ways to collect feedback on their services and image. This in turn leads to the improvement of their quality. Nevertheless, tourists can take informed decisions with significantly less effort, due to the wide access to information that is available online.

In [47] the authors introduced the term of smart tourism to describe "the application of information and communication technologies to the tourism section". A plethora of smart tourism portals are available globally providing different functionalities such as: post-visit experiences sharing, description of tourist attractions, highlights and advises, attraction recommendations, geo-tagged photos,

etc. Moreover, such portals address various touristic aspects, like for example: historical places, landscape, sightseeing, food, restaurants, shopping, entertainment, weather, etc. Hence, we can claim that the world of smart tourism is worth investigating for gathering its better understanding that might lead to further development of the field.

As object of study, we have chosen the portal *AmFostAcolo*[1], which is one of the most popular smart tourism portals in Romania. This portal allows registered users to exchange knowledge by posting impressions and questions. At its core, the portal provides forum like functionalities and targets Romanian language speakers that search touristic information about national and international touristic entities.

Throughout the section, we shall use the term touristic entity to designate a geographical area or geographical attraction of touristic interest. There are no limitations on the area size of an entity, e.g., a touristic entity can refer to Italy as a country, Athens as a city or to *Manneken Pis* as an attraction. Furthermore, there is no limitation on the touristic entity as it could refer to a city, a cliff, a field of tulips, a hotel, an entertainment park, a statue, etc.

This chapter is based on the following work, presented in previous papers:

- In [12] the data extraction mechanism was presented and some preliminary analysis were conducted.
- In [27] and [28] we presented results based on conduction CNA analysis on a touristic content sharing network.
- In [26] a new keyword extraction mechanism from touristic content sharing was proposed.
- In [23] we investigated the traits of the communities involved in touristic content sharing.

We structured this chapter as follows. Section 2 addresses smart tourism's background based on relevant research papers. The following sections briefly introduce the field of network science and discuss its possible applications in tourism. Section 4 briefly introduces the field of natural language processing and its use in

[1] http://amfostacolo.ro/

touristic applications. Next, we discuss about graph databases and their advantages, also mentioning projects sharing similar goals with our own goals. Section 6 describes the experimental scenarios held, while section 7 discusses the obtained results. Last, we state our conclusions and present possible future work.

2.1 Background & Related Work

2.1.1 Smart tourism

In the early days of the *World Wide Web*, Poon A. [176] emphasised that information and communication technologies (ICT) will shape the ways of doing tourism. The author claimed that a new type of tourism was emerging driven by "new consumers, new technologies, new production practices, new management techniques and changes in the industry frame condition". According to Carter & Bédard [50], the *World Tourism Organisation* recognised that ICT was already playing a key role in the entire tourism industry and it published a guideline for the adoption of ICT by destinations and business. By the year 2008, Buhalis & Law [45] claimed and provided evidence that "the technological revolution experienced through the development of internet has changed dramatically the market conditions for tourism organisations". Moreover, the authors emphasised that ICT technologies have empowered a new type of traveller that is "seeking exceptional value for money and time" and is "keener to pursue their own preferences and schedule". An example of a technological empowered accommodation business is Airbnb[2], it has become the largest hotel chain in the world without actually owning a hotel. This type of business has pushed previous industry leaders, e.g. Marriott, to invest heavily in ICT solutions. Furthermore, Airbnb's impact has been noteworthy even for large touristic destinations, such as the city of Barcelona in Spain, see paper [104]. This work has shown that Airbnb's accommodation created a clear demographic pressure in the city. Furthermore, the results "suggest that Airbnb benefits in greater measure than hotels from proximity of the most visited places in the city".

Later, the researchers have set more focus on developing Smart Tourism. In particular, the development of recommender systems, as well as the use of mo-

[2] https://www.airbnb.com/

bile technologies has attracted a lot of scientific work. Borràs J. [37] has identified the main advances in the use of recommender systems for tourism. Rodriguez-Sanchez, M. C. et al. [188] and Gavalas D. et al. [93] have proposed and presented relevant mobile technologies for Smart Tourism. Semantic methods were also used to develop Smart Tourism functionalities, see papers [3] and [90]. Among other research projects, we mention Jiang K et al. [121] effort to use geo-tagged photos in ranking personalised tourism attractions, as well as investigations in the use of social media applications by Gretzel U. in [98] and No E. & Kim, J. K. in citeno2015comparing. Sigala, M. & Chalkiti K., see paper [201], investigated the use of Web 2.0. by Greek touristic companies. Although many companies used content sharing networks, few of them effectively exploited this new opportunity for knowledge management activities. Hence, a contribution of our study is to provide an innovative analysis of a touristic content sharing portal, aiming to get a better insight into the potential, opportunities and problems of employing such a business model in the tourism industry.

2.1.2 Natural language processing

Texts expressed in natural language are the most common way by which consumers of touristic services present information on Smart Tourism portals. These texts describe users' opinions and/ or experiences on various touristic entities. Hence, a need arises to analyse these texts using natural language processing (NLP) methods.

NLP is an applied computing field interested in the automated computational analysis of natural language texts (Manning C. D. & Schütze H. in [145]). Jackson, P. & Moulinier I. in [118] present a survey of the use of NLP for online applications. Natural language texts are composed of sentences, hence various linguistic theories were proposed along the ages to understand natural language sentences. They all agree that the foundation of sentence structure relies on identifying and analysing the various relations among words (subject, complement), see Hristea F. T. in [115]. The most common NLP approach of representing these relations is the dependency graph, encapsulating both syntactic and semantic features. Dependency parsers are used to construct dependency graphs, based on the theoret-

ical foundations of dependency grammars, see Kübler S. et al. in [130]. Hence, graphs represent a natural approach to capture and process connections between words in natural language texts. They were proven to be efficient for lexical acquisition by Widdows D. & Dorow B. [229], sentence parsing by Radev D. R. & Mihalcea R. [185], in word sense disambiguation Navigli R. & Lapata M. [161], or text summarising by Mihalcea, R. [154].

A series of research papers supports the utility of NLP within the context of Smart Tourism. Yuan. H et al. in [235] proposed a method to detect travel routes by determining co-occurrences of travel destinations extracted from texts posted on travel blogs, aiming to provide tourists with better travel scheduling. Through aspect-based opinion mining on TripAdvisor [3] Smart Tourism Portal, Marrese-Taylor E. et al. [146] were effective in determining the sentiment orientation of opinions. García-Pablos A. et al. in [91] introduced a platform (OpeNER) applied to the hospitality domain that processes customer-generated content to obtain valuable information. In another research paper, see paper [80], Filieri R. & McLeay F. identified strong predictors for the travellers' adoption of information, from online expressed touristic reviews. Hence, travellers could be informed upfront on the quality of the reviews to further simplify their decision making process.

In this study, we rely on NLP methods such as sentence segmentation, tokenization, Part-Of-Speech tagging (POS) and lemmatization. More details of these methods are presented in [202]. For bench-marking regarding the accuracy of our proposed approaches, we employ standard measures, like for example Term Frequency Inverse Document Frequency (*tf-idf*), introduced in [206], and initially proposed in the field of information retrieval.

2.1.3 CNA in Smart Tourism

CNA was also used in the context of Smart Tourism. Some of the works, e.g., Bendle L. J. & Patterson I. [30], Miguéns J. I. L. & Mendes J. F. F. [153] and Presenza A. & Cipollina M. [183], analysed only the topological aspects of the networks constructed from touristic oriented sources. Results confirm the complex nature

[3] https://www.tripadvisor.com/

of the networks as they have a power-law degree distribution. From a business point of view, some studies were focused on identifying the most relevant actors involved in using and supporting various touristic destinations around the world, see the study on the Elba region in Italy by Baggio R. & Cooper C. in [14] and Waitomo Caves in New Zealand by Pavlovich, K. in [175]. Tourists mobility represents another direction of CNA application. For example, Taczanowska K. [209] focused on understanding the structure and the use of hiking trails. More elaborated details on the applications of CNA in Smart Tourism are presented in [13] and [220]. As the latter paper emphasises, there is still a lot of work to do for proving the value of the use of CNA in Smart Tourism, especially from the tourists' point of view. Hence, our study will try to provide knowledge that will ultimately be in the benefit of the tourists.

There are many software tools for supporting NS applications, including toolkits, as well as libraries and APIs. In our NS analysis, we utilised *Gephi*, a tool developed by Bastian M. et al. [19] with the general purpose of "visualisation and exploration for all kinds of graphs and networks". Also, we employed Python's NetworkX package for exploration and analysis of networks developed by Hagberg A. et al. [105]. These tools as well as our experiments rely heavily on graph theory / CNA metrics such as *Degree*, *Average Degree*, *Diameter*, *PageRank*, *Modularity*, *Average Clustering Coefficient*, *Average Path Length*, *Diameter* and similar others, which will be introduced in the Discussion and Results section. Boldi P. & Vigna S. [34], Page L. et al. [173], and Blondel V. D. [32] present in-depth details on these metrics. The presence of social phenomena such as: i) the small-world phenomenon described by Travers J. & Milgram S. [215]; ii) rich-get-richer/power-law degree distribution phenomenon overviewed by Adamic L. A. & Huberman B. A. [1]; iii) and preferential attachment phenomenon discussed by Newman, M. E. [163], are of great interest for our study. They provide crucial insight about the dynamics of social networks, as well as the behaviour of people involved.

2.1.4 Graph databases

"Graph databases leverage complex and dynamic relations in highly connected data to generate insight", and "they represent the best way to represent and query

connected data" as stated by Robinson I. in [187]. Since our work is centred on discovering textual relations from reviews and touristic social relations, the choice of using graph databases comes natural. Moreover, this type of database has the advantage of being readily available for any join-like navigation operation, that we will heavily rely on in discovering information, as opposed to SQL and NoSQL database systems. As tool, we used Neo4j [4], as it is an ACID compliant transactional database with native graph storage and processing. Nevertheless, Neo4j has also been proposed in other Smart Tourism research papers. For example, Karakostas, B. et al. [124] have employed Neo4j to develop a knowledge-based system for travel mode recommendation, while Jinyan, C. et al. [122] explored tourist's observations for monitoring environmental, with the help of Neo4j.

2.2 Experimental scenarios

2.2.1 Data Source

As data source we use the information available on Romanian website *AmFostAcolo* (English translation is "I have been there"), which is one of the most popular tourist opinion sharing online portals in Romania. This portal provides semi-structured data oriented on post-visit tourist reviews about a large variety of locations covering specific aspects of accommodation as well as general impressions about touristic sights. From a geographical point of view, tourist attractions depicted on the website mainly cover Romania and Europe with less data for other continents. This is due to the visitation habits and financial restrictions of native Romanian tourists. The data is hierarchically structured according to the country, region, section and geographical place. For example, *Corte Zeuli B&B* is a place under the country *Italy*, region *Puglia* and section *Bari*.

A potential user of the portal can interact with other users in two ways. First, the user can ask questions and provide answers to other questions. This functionality is especially useful for planning purposes, before travelling to a specific location for which enough suitable survey information is not available. Secondly, users can post tourist impressions related to certain places. Hence, for each place, a mul-

[4] https://neo4j.com/

titude of impressions posted by various users is available. Moreover, impressions can be commented using echoes issued by other users, with the goal either of presenting further clarifications or of requesting additional information. Although echoes and questions might apparently share similar functionalities, echoes can only exist in the presence of an impression, whereas questions can be posted unrelated to any impression. For a better understanding of the structure of the portal, please consult Figure 2.2.1. The portal automatically associates each user with a rank computed based on the user portfolio and feedback provided on the website, including facts like: impressions, uploaded photos, posts, and answers.

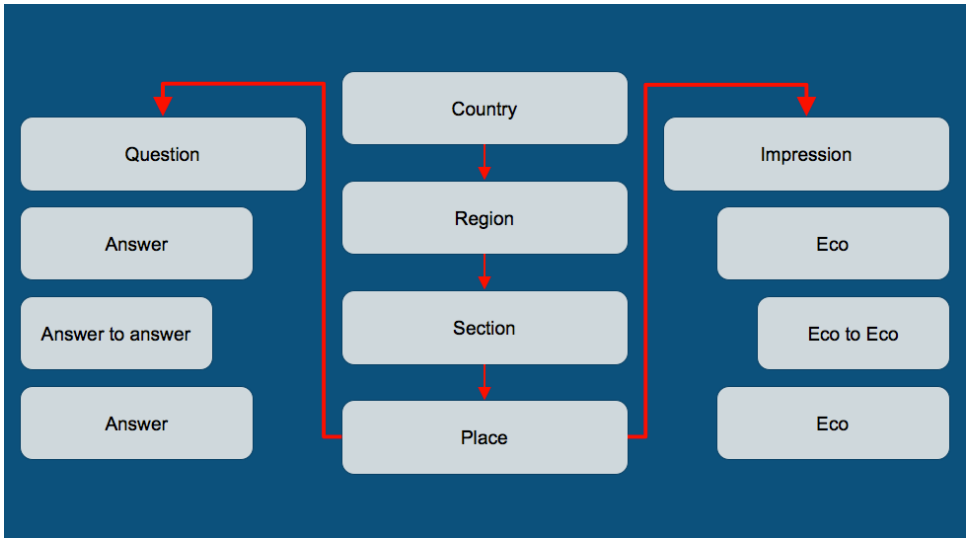


Figure 2.2.1: *AmFostAcolo* structure and interaction functionalities.

2.2.2 Question-answering scenarios

Our first scenario focuses on the analysis of the *AmFostAcolo* users' social network that emerges from using the question-answering functionality. We focus on this functionality for inferring the social network describing users' interaction, as apparently more users have taken advantage of it rather than the impression-echoing functionality, i.e. 53172 questions versus 2541 impressions. This difference can be explained as follows. We claim that using the question-answering functionality requires less effort, i.e. reading and understanding of shorter texts, than longer

impressions. Also, it is assumed that a user posting an impression on a touristic attraction has already visited it, although the portal does not enforce any kind of verification. Hence, users are limited in posting impressions versus posting questions. Apart from analysing the resulting social network, we also used Neo4j to link users to questions and answers, with the purpose of developing a method for digesting users' interests.

The first step was to extract the data from the portal using web scrapping, more specifically by applying Java's jsoup HTML parsing library[5]. A series of XML files were obtained, one for each user, with the XML schema detailed in Figure 2.2.2. For experimental purposes and ease of manipulation, a MySQL[6] database was populated with the data present in the XML files (see Figure 2.2.3 for the relational schema). Thereafter, we could count 54352 users out of which 3132 have asked at least a question and 7831 that gave at least an answer. In total 8936 (active) users have been using the question-answering functionality. Also, we counted 53172 questions and 53150 answers. If we take into consideration only the users that actually posted questions, we get an average of 16.1 questions such user, respectively those that posted answers, we get an average of 6.7 answers per user. Furthermore, we exported the database into CSV files, as an intermediary step for importing the data into Neo4j. Also, by using MySQL we could easily import the data into *Gephi* network analysis tool. For a better understanding of the entire extraction and preprocessing step, please consult the graphical representation shown in Figure 2.2.4.

Graph Database Experiments

Graph data models consist of nodes and arcs, where nodes represent entities and arcs capture binary relations between entities. Both nodes and arcs can be enriched with attributes, e.g. if a node represents a user we can attach the age attribute. In this scenario, we model three types of entities: user, answer and question. We considered three relations in our model. Relation r : *ASK* connects a user (qUser) and a question. Relation r:HAS links a question and an answer. Relation r : $GIVEN_BY$ is defined between an answer and a user providing that answer. You can inspect the corresponding Neo4j data model in Figure 2.2.5. 2.2.5. For

[5] https://jsoup.org/
[6] https://www.mysql.com/

querying purposes, we employed Cypher, the native querying language for Neo4j. McKnight W. provides more details on Cypher in [148].

With the help of *AmFostAcolo*'s question-answering data available in Neo4j format, we were able to identify the interests of a user, an operation known as user profiling. Consider the following scenario: users A, B, and C are a group friends who decided that they should spend their next vacation together. Nevertheless, each user has its particularities and interests and they all would like to go to a destination that pleases each of them, a taunting task for the travel agency that they contacted. Having access to their previous question-answering data, the travel company can develop a profile for each of them by consistently combining their interests in order to provide them better recommendations.

To further prove the utility of using Neo4j with the data gathered from the touristic portal consider the following scenario. User A is looking for users that have similar interest, aiming to discover potential new (for user A) touristic locations to visit. Hence, we could compute similarity among users taking in consideration the percentage of questions answered, questions posted by user A, by other users. For example, if user A posted 10 questions out of which 3 were answered by user B and 5 by user C. Then, user A has a similarity of 30% to user B and 50% to user C. In order to answer a user's question another user has to visit the place X on the portal, where the question was addressed. Hence, we can deduce that both users have an interest in place X, from which we deduce the similarity. This scenario can be thought as recommender system for users with similar interests. As we will further prove, through Neo4j we can easily build this recommender system.

CNA scenarios

For this part of the study, our entities of interest are the users, represented by nodes of the social graph. We created arcs as follows: an arc connects source node A to destination node B if node B posted a question upon which node A answered; A is the source node and B the target node. We have considered only the socially engaged users, i.e. users that posted at least one question or one answer. Thus,

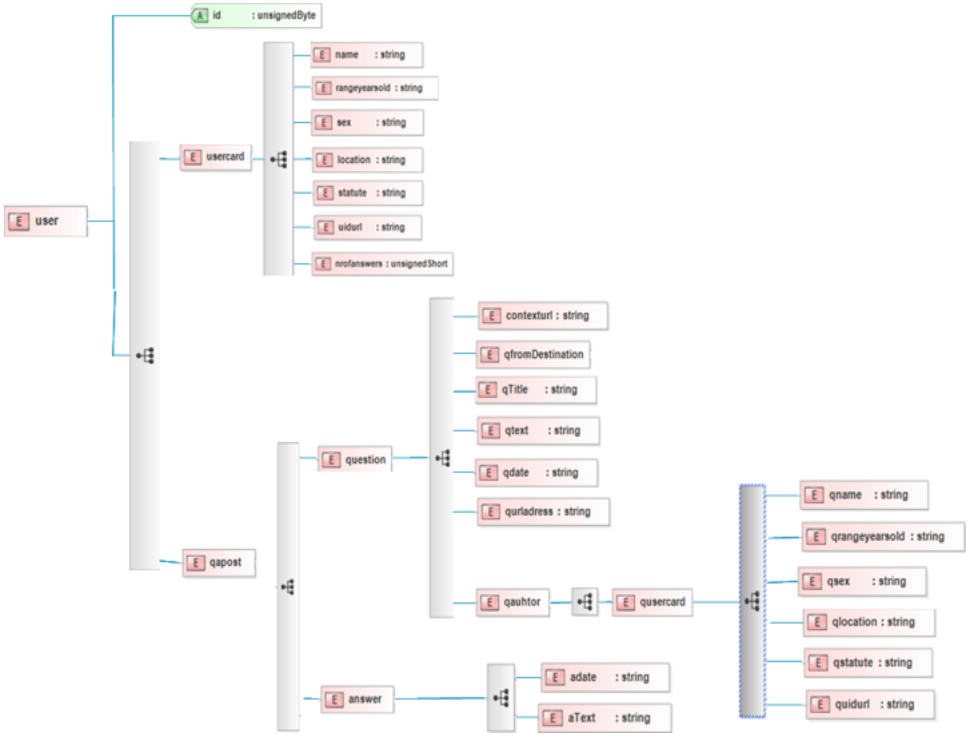


Figure 2.2.2: XML schema capturing user interaction data.

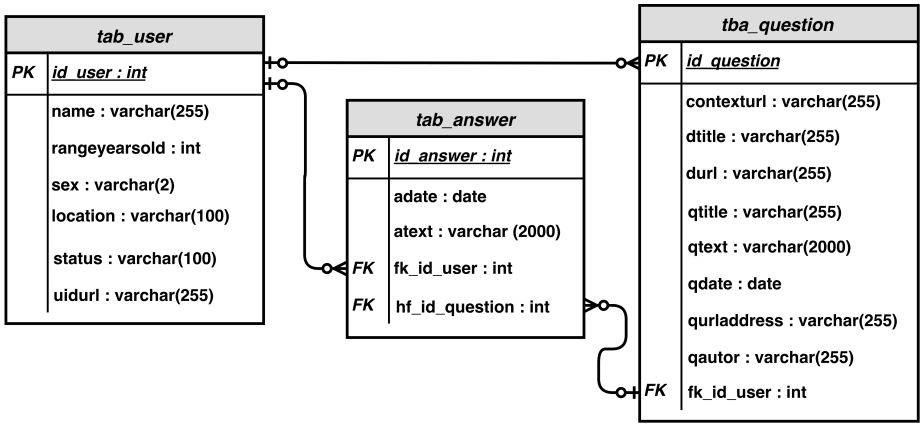


Figure 2.2.3: Relational schema describing the extracted data on users, questions and answers.

we created a social network based on the question-answering functionality of the website. This network contains 8936 nodes and 29529 arcs. Further, we eliminated self-looping arcs, as they represent answers posted by a user to its own question and

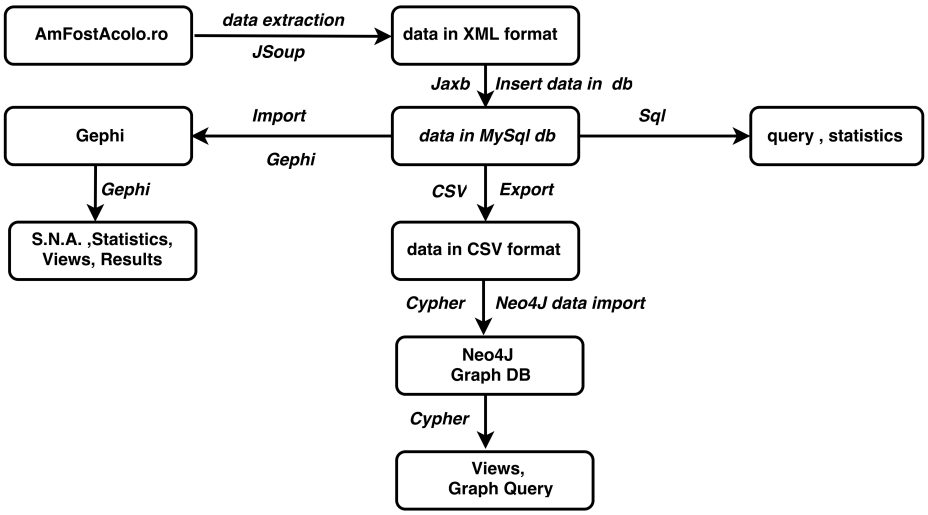


Figure 2.2.4: Relational schema of the extracted data.

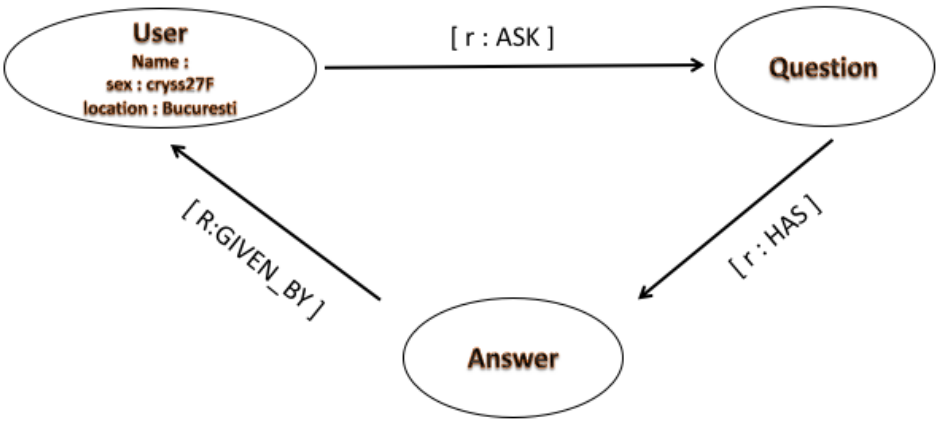


Figure 2.2.5: Neo4j data model of users engaged in question-answering interactions.

those users that answered only to their own questions. Consequently, we reduced our network to 8017 nodes and 25666 arcs. Moreover, we annotated nodes and arcs with their creation date, to enable temporal analysis. Our current experiments follow three directions:

- Inspect the topological aspects of the network, in order to understand how users socialise.

- Provide a temporal analysis, in order to understand the evolution of the social network.
- Determine inner communities of users and their traits, to identify trends and similitudes among users

We focused on the following questions that we consider of high interest:

1. Is our network a complex network and if so what type of complex network? What are the traits of the complex network type?
2. Are there present any sociological phenomena and if so what are their traits?
3. Is the network resilient?
4. How did the network evolve over time?
5. Is the network expanding or contracting?
6. Is there any evidence that a specific positive or negative review had or may have significant influence on the community?
7. Can we determine inner-communities and their specific traits?

2.2.3 Impression-echoing scenarios

This experimental scenario follows the *AmFostAcolo*'s impression-echoing functionality. Using the web scrapping technology, we were able to collect impressions posted by 1085 users (423 males and 662 females), totalling a number of 2521 impressions, dispersed over:

- 45 countries
- 161 regions among which 16 country sub-regions and 145 other regions
- 529 sections among which 489 localities (cities, town or villages)
- 1420 tourist locations among which:

- 534 accommodation units (cottages, pensions, hotels, houses or villas)
- 886 sections, i.e. representing impressions about a tourist location

In our current scenario, the goal is to define new text analytics methods for analysing tourist impressions/reviews based on NLP and CNA methods, aiming to better capture users' touristic interests. The proposed method is fully unsupervised, see Figure 2.2.6 for a graphical representation, and proceeds as follows. For each review, we captured the title, text and metadata. The title and the text were parsed through a chain of NLP methods that include: tokenization, Part-Of-Speech tagging and lemmatization. Hence, the boundaries of each sentence were marked and the part of speech and lemma were attached to each word, in order to unify the representation of all occurrences of the same lexical unit. Next, the review title and content were fully annotated with dependency relations, using the *Romanian Dependency Parser*[7]. A graph/complex network (undirected graph) is resulted, nodes representing lemmas of words and links for dependencies, while keeping the dependency names as link attributes. We used Syntactic filters to remove nodes out of the graph, only keeping nodes with meaningful information. For example, we could consider only nouns and verbs as being meaningful, as carrying relevant semantic information. Hence, we could just keep nodes that represent nouns and verbs and their respective links.

For this experiment, we decided to consider only the words that have stable lexical meanings, i.e. nouns, verbs, adjectives and adverbs. We ignored function words, i.e. prepositions, articles and pronoun, as they only have syntactic functions, i.e. they do not convey valuable information for our analysis. Next, we fusion the different occurrences of the same word in a unique node and then we gather all relations in which each occurrence was involved. The syntactic filtering and unification steps of different occurrences reduced the size of the complex network, making its computational analysis easier, often resulting in a sparse graph/complex network. This method of representing text from reviews/impressions entirely covers all the dependencies, i.e. there was no loss of valuable information. Our analysis forked into two directions: text summarization using keyword extraction

[7] http://nlptools.infoiasi.ro/WebFdgRo/

via *PageRank* algorithm and words' cohesion checking.

The first direction assumes the review summarization by identifying relevant words as keywords. We achieved this by computing two different NS metrics for each node: *PageRank* coefficient and *Degree*, and then ranking the nodes according to the metric used. According to *PageRank* metric, a word is a keyword if it is highly linked with words that are also highly linked with other words. According to *Degree* metric, a word is considered a keyword if is linked with many other words, with no regards on the others linkage. In order to evaluate the proposed keyword identification methods, we investigate the occurrences of keywords among title words and metadata. The review metadata consists of location information, i.e. country, region, section, and location name. The metadata was also included in this analysis since no words in the title were present in the review's text for 75% of the reviews. This evaluation assumes that discovered keywords and words in the title represent the summary of the review. Moreover, we also used a third method based on the standard tf-idf from information retrieval, to compute the keywords capturing the semantics of each review. We computed the term frequency for each separate review, while the inverse document frequency was computed for the entire corpus of review texts. We introduced this third method in order to benchmark our proposed methods versus a standard and widely used keyword extraction method.

The second direction tries to determine if the words defining each review have high cohesion. This would imply that the expressed ideas are tightly connected. This metric could be used to judge if the reviews were not randomly generated, and thus increasing the relevance of our experimental results. We assume that textual cohesion is present in texts if the small-world phenomenon is observed in the corresponding graph representations of those texts. This in turn is depicted by evaluating the following CNA metrics: *Average Clustering Coefficient, Average Path Length* and *Diameter*. Moreover, we would expect this to correlate positively with the presence of a giant component in the graph that represents the text.

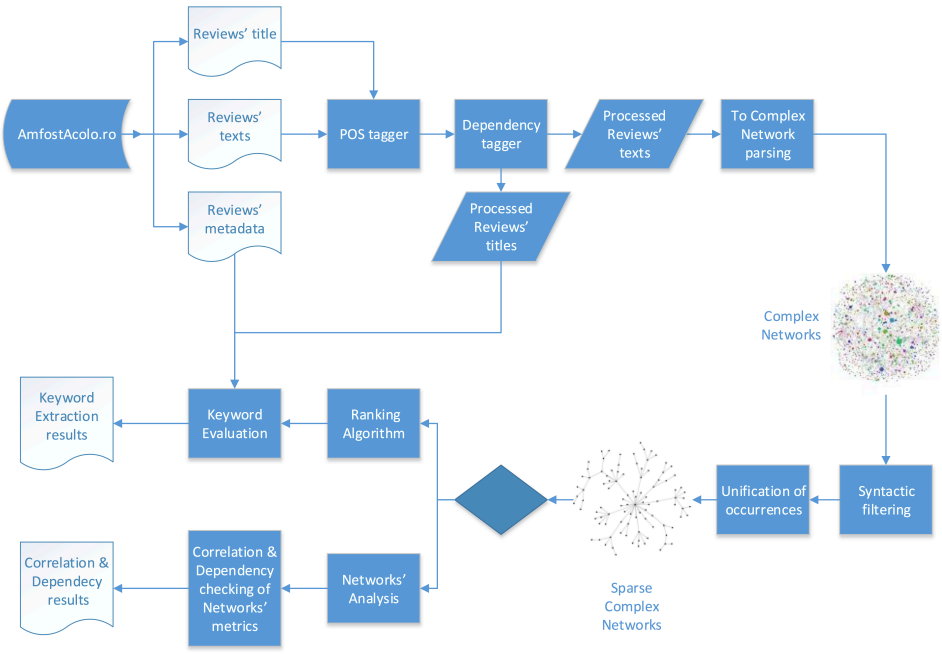


Figure 2.2.6: Impression-echoing experiments pipeline.

2.3 Results & Discussions

Following the introduction of the experimental details, we are now discussing the results obtained. We have aligned the presentation of the results with the description of each experiment.

2.3.1 On Question-Answer Scenarios

Graph Database Experiments

The availability of *AmFostAcolo*'s question-answering data in Neo4j format helps in identifying the interests of each user (profiling). As for the first scenario regarding graph databases, a travel agency can use Cypher extract the questions and answers of a given user respectively, as depicted in Listing 2.1.

```
MATCH (uq:User)-[Ask]->(q:Questions)
    WHERE uq.userId = "15"
    RETURN uq,q
MATCH (uq:User)-[ASK]->(q:Questions) )-[h:HAVE]->
```

```
    (a:Answers) - [r:GIVEN_BY] -> [ua:USER]
  WHERE ua.userId = "15"
  RETURN ua,a,q
```

Listing 2.1: Cypher User profiling query.

The second graph database scenario, regarding the recommendation of similar users, can be fulfilled by using the Cypher query language to determine the subgraph of users that gave answers to questions asked by a certain user, as in Listing 2.2.

```
MATCH (uq:User)-[Ask]->(q:Questions)-[h:HAVE]->
    (a:Answers) - [r:GIVEN_BY] -> [ua:USER]
  WHERE uq.userId = "15"
  RETURN uq,q,a,ua
```

Listing 2.2: Cypher user recommandation query.

We acknowledge the fact that many more scenarios can be envisioned and that the data model can be extended. However, we claim that these experiments have proven the usefulness of graph databases in the context of Smart Tourism.

CNA Experiments

We start our network science experiments by focusing on the topological aspects supported by various NS metrics. The *Diameter* metric represents the size of the longest path connecting two nodes of the graph, with a computed value of 16 for a graph of 25666 we can argue that observed value is quite low. Furthermore, the *Average Path Length* is 5.042 which we consider to be low for our graph. Both the *Diameter* and the *Average Path Length* are predictors of the small world social phenomenon, which we argue to be present in our network due to the relative small values of the metrics in question, see paper [215]. The implications of the small world presence indicate a tighten community of users, where information travels fast. Hence, we can argue that either positive or negative information on a touristic attraction has a significant influence on the community, as it can spread fast and easily from user to user, thus answering question six. Furthermore, additional empirical tests on the diffusion of information strengthen the evidence on the presence of the small world phenomenon. In the diffusion experiments, we chose as

sources the nodes at the periphery of the network, with a diffusion loss of 70% at each step, assuming that only the neighbours and neighbours of neighbours of the source node could further broadcast. The results have shown that a significant or even large majority of the nodes will receive some information. Figure 2.3.1 presents a graphical exemplification of the results of the diffusion experiment.

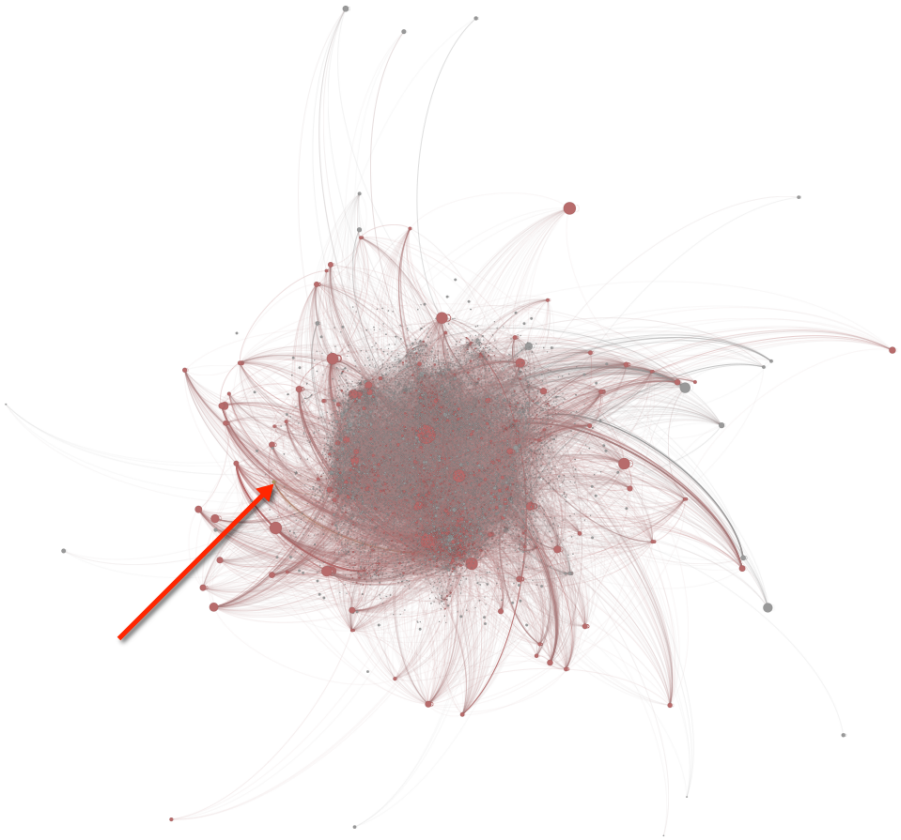


Figure 2.3.1: *AmFostAcolo*'s social network with diffusion experiment. The nodes' diameters are proportional to their *PageRank* coefficient. The colour of the nodes depicts if information through the diffusion experiment has reached them (red) or not (grey). For plotting we used *ForceAtlas 2* algorithm included in *Gephi*.

The *Average Degree* has a value of 3.201, while the *Average Weighted Degree* has a value of 5.881. This means that in average a user responds to 6 questions addressed

by 3 other users. Hence, the average social activity is low, which also corroborates with the value of the *Average Clustering Coefficient* of 0.015. On computing the distribution of the nodes' degree, we observed that the distribution has *scale-free* type. This implies that few users are very active (acting as hubs) while the great majority are not, which is in line with the values obtained for *Degree* and *Average Weighted Degree* metrics. Moreover, this distribution occurs frequently in the Internet, as well as in the natural world, as stated in [1] and other Smart Tourism complex networks as mentioned by Baggio, R. [13]. According to the statistical mechanics of scale-free networks studied by Barabási A. L. et al. in [18], this type of networks emerges if both growth and preferential attachment social phenomena are present. Hence, we found evidence that the social network is growing, answering question 5. Newman M. E. defined the preferential attachment phenomenon in [163] as a set of processes in which some quantity, quality, or typically some form of wealth or credit is distributed among a number of individuals or objects according to how much they already have, so those who are already wealthy will receive more than those who are not. This phenomenon is also known as rich get richer. In our context this implies that posts from a very active user (responded to many questions) weighs more (as is more trusted) than an answer from a typical user. The plot presented in Figure 2.3.2 shows that the nodes' degree follows a scale-free distribution. These results presented on the sociological phenomena provide answers to questions 2 and 4.

According to Figure 2.3.1, the visual perception suggests that this network is of the type core-periphery, as described by Hojman D. A. & Szeidl A. in [110]. This type of network has a central conglomerate of very well connected (core) nodes, while the outer parts of the network (periphery) are more scattered. To prove the validity of this empirical observation we need to extract the core and analyse its connectivity. Since the core should be the higher connected part of the graph, we could extract it by removing the nodes with a degree less than 30 (10 times the average) and their respective links. Hence, a graph of 300 nodes (3.74% of the total) and 2080 (8.71% of the total links) was obtained (core graph). Then, we computed the following CNA metrics and compared the results with those attained for the entire network. Unsurprisingly, the results indicate a higher connectivity of the

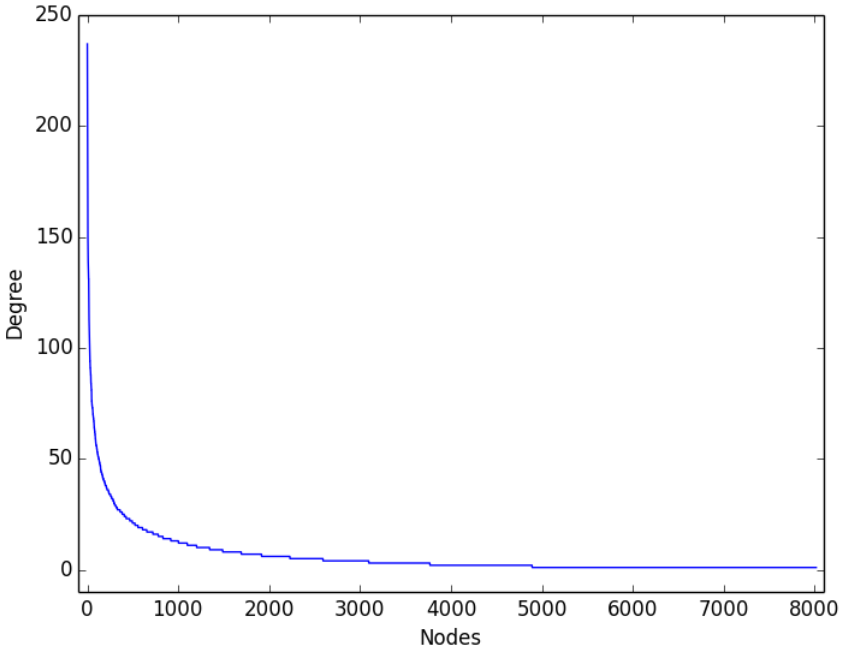


Figure 2.3.2: *AmFostAcolo*'s distribution of the nodes' degree for the social network.

core graph. The *Average Degree* is 10.267 versus 3.201, while the *Average Weighted Degree* is 28.477 versus 5.881. These numbers indicate a significantly higher interaction activity with more users, respectively more questions answered in average by the users in the core. The *Diameter* value is 9 versus 16 and the *Average Path Length* value of 3.262 versus 5.042, implying a tighter connected graph. This also corroborates with the *Modularity* coefficient of 0.299 versus 0.48. Nevertheless, the *Average Clustering Coefficient* depicts a connectivity that is 4 times larger in the case of the core than on the entire network, 0.068 versus 0.015. To become part of the core members, one has to be active, hence in a sense we can argue that the social network is meritocratic and can be trusted. The core plays the role of a diffusion enhancer, which is consistent with the rest of our findings. Based on the above findings, we can sustain that the entire network is of *core-periphery* type. A relevant trait for this type of complex network is its resilience to change. This implies that the giant component of the network will be divided only in the case of a

catastrophic event, e.g. the loss of a significant part of the most active users. Hence, we consider answered both question 1 and 3.

Next, we focus our topological analysis on the inner communities that rise in the social network (further called communities), thus lowering the granularity level of the network. For the detection of these communities, we used the *Modularity* algorithm devised by Blondel, V. D et al. in [32]. First we compute the *Modularity* coefficient which lies in the range of $[-0.5, 1]$. A positive value of the coefficient suggests that the number of links within groups exceeds the number expected on the basis of chance, i.e. randomly assigned links. For our network, the value of the *Modularity* coefficient is 0.48. This supports the presence of social communities. The number of detected communities is 146. However, this large number is biased by the presence of small and/or peripheral communities. Hence, we considered only the communities that are part of the graph's giant component. There are 25 such communities. In order to analyse the interactions among these communities, we constructed a directed graph of communities (let's call it Gco) as follows. Nodes of the same community fused into a single node, the community node. Links among members of the same community disappeared, while links among members of different communities fused into single links among the representative community nodes' of each user. Based on the initial data from the portal we added directionality to the links, such that the source node is the community node representing the user answering the question and the target is the node representing the community of the user that posted the question. Upon computing the *Average Degree* on the Gco we got a value of 47.2, see Figure 2.3.3 for more details. We computed the *Degree* as the sum of the *In-Degree* and the *Out-Degree*; since we have 25 nodes, the maximum value for *In-Degree* and *Out-Degree* is 24. Thus, the maximum value for *Degree* is 48. This implies that users from different communities do interact, which further strengthens the finding that information can travel fast and with ease. Figure 2.3.4 presents the graph of communities.

Since we were able to detect inner-communities in our network, we claim that we have partially answered question 7. To consider this question fully answered, we still have to determine and point out touristic traits that are specific for each

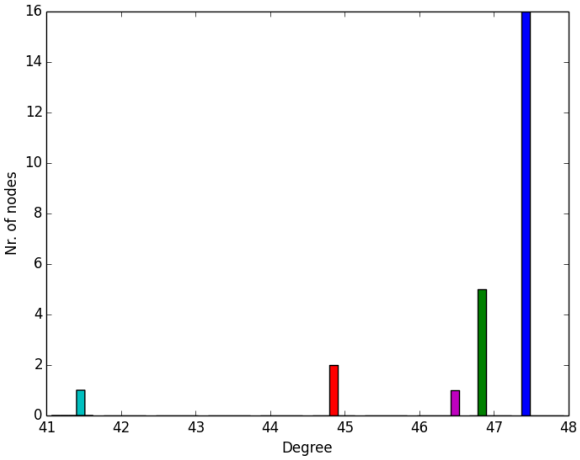


Figure 2.3.3: *AmFostAcolo*'s *Degree* distribution for nodes in the graph of communities (Gco).

Figure 2.3.4: *AmFostAcolo*'s community graph. Each node represents a community; nodes are distinguished by their unique color. The diameter of each node is proportional to its *PageRank* coefficient.

community. In this endeavour, we also used the *Modularity* algorithm to detect the communities, but we did not restrict its application only to those communities that are part of the network's giant component. In order to capture only the relevant communities, we filtered out those with less than 10 users. We propose two approaches for discovering traits based on the available data, captured from the portal. Concerning users' metadata, we can compute the average scores per

community for the following: sex, age, average score given per touristic entity, number of sentences written and rank (computed by the portal). If these averages vary from community to community, then we can assume that they represent specific traits of each community. Since we have obtained the texts for questions and answers for each user, we propose to aggregate these texts per each community and then determine textual traits for each resulting text. First we parsed each users' questions and answers using the Part-of-Speech tagger for Romanian language described in [114], thus obtaining a set of tokens together with their occurrence frequency. Next, we aggregated the tokens for each community of users and computed their community frequency. We filtered tokens to keep only those tokens representing nouns, verbs and adjectives, as the rest of the tokens possess only syntactic information that is not relevant for our purpose. Moreover, we ranked decreasingly by their occurrence frequency the nouns, verbs and adjectives and considered the top 10 most frequent as candidates for capturing touristic traits. Hence, for each community we obtained 3 lists of the 10 most frequent tokens for the parts of speech considered.

Regarding the metadata traits, we were able to reach some results. The users' age varies between 30 and 36 years old and the users' rank varies between 3.8 to 5.4 (on a scale from 0 to 12), for each community. Due to their low variance, these parameters cannot be considered as representing specific traits, while the users are apparently uniformly distributed. Unfortunately, the results obtained for the other metadata (sex, average score given per touristic entity and number of sentences written) were not significant, so this aspect needs to be further examined in future works. However, the communities have a larger variance for noun and verb textual tokens. Hence, we argue that the top most frequent nouns and verbs can be seen as specific touristic traits for each community. As a fact, we observed that the verb to eat is the only frequently used verb for the vast majority of communities. Hence, we can argue that eating is considered a key activity regardless of tourists various interests. Despite having also a large variance, the large majority of adjectives refer to monetary values, which do not provide useful information by themselves. Here the context plays a key role that we did not properly consider in this experiment. Hence, we claim that further studies should consider nouns and verbs as specific traits for each community, while adjectives should be captured together with some

context information.

Our topological study could continue by further lowering the granularity of the analysis towards nodes. However, we left this kind of deeper investigation as future work. Nevertheless, we identified two directions of interest. The first one should focus on the nodes importance in the social network. This can be determined using the *PageRank* algorithm or using the *Betwenness* metric to highlight the users that play a major role in the spread of information. Rewarding (given by the portal) the users of great social importance may represent the necessary incentive for those users to further continue and/or improve their presence on the portal. Thus, the portal, the users in question and the other users will see the benefits of a more "social" network. Another direction involves using the geographical metadata for the users and questions, to determine touristic trends of visitation. It is possible to create a complex network where nodes represent locations and links represent interest in locations that can be further analysed as described in [153].

As initially planned, we also conducted a temporal analysis of the social network. In order to fulfil the analysis, we retrieved from *AmFostAcolo* portal the date of each question and answer posting. As methodology, we considered capturing a yearly snapshot, from 2010 to 2015 on the 1st of January of each year. Previous data were also available, but the network was of negligible size. As Figure 2.3.5 shows, the network continuously grew in size and developed its shape, from the core to the periphery (like a tree). The growth process appears to be cyclical as follows:

1. Expansion step. It consists in attracting new users in the network. Observe the years 2010 and 2014.

2. Development step. It consists in new links creation among the current users with less attraction of new users. Observe the years 2011, 2012, 2013 and 2015.

3. Return to step 1.

Hence, the cycle appears to span a number of years, with the first year being the expansion step, while the following 3 years are devoted to the development step. However, we believe that this temporal analysis must be repeated for the following

years (2016 and afterwards) for strengthening this analysis. Nevertheless, we have yet again brought evidence that the social network is expanding, thus providing an answer to question 5.

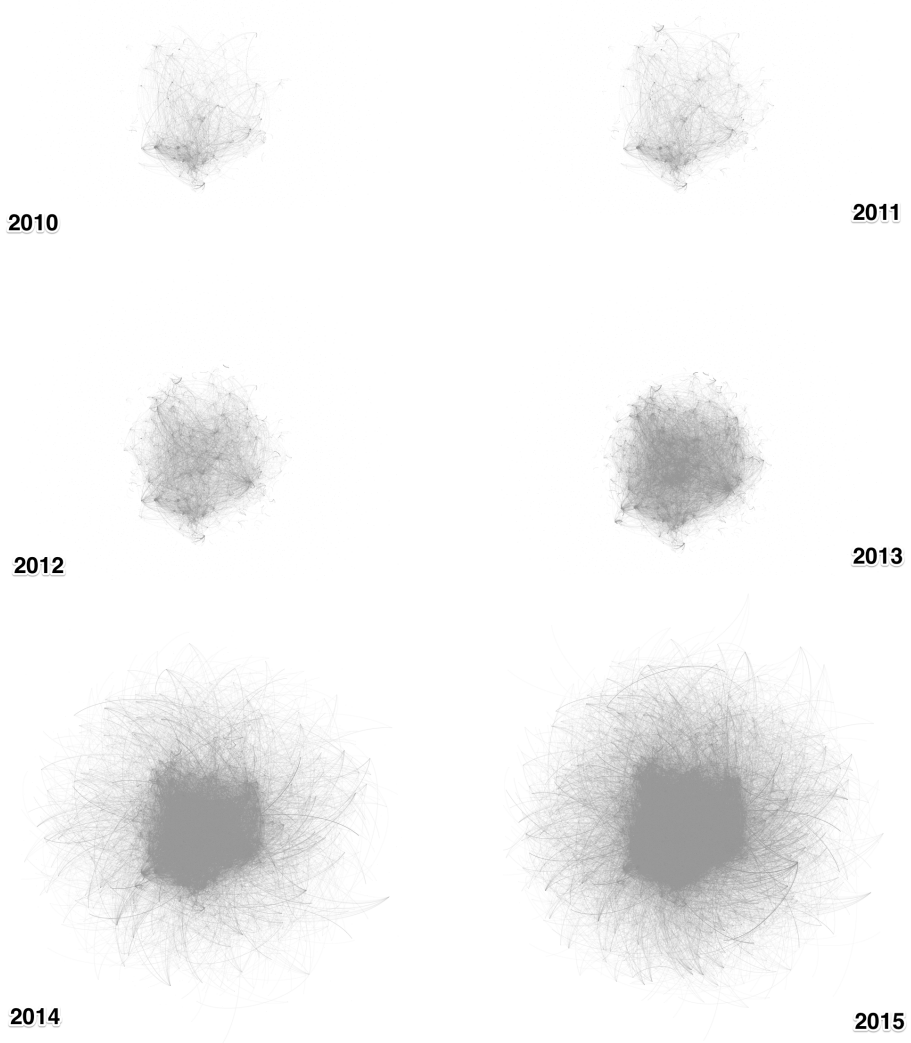


Figure 2.3.5: *AmFostAcolo*'s time lapse of the social network as captured of the 1st of January of each year.

Table 2.3.1 shows a summary of the results for the keyword extraction process. Before commenting on the results, we should mention that just for 723 reviews (out of 2521) we could find a common word in the title and in the text of the review, while considering nouns, adjectives, adverbs, and verbs. Hence the inclusion of the metadata is fully justified as it leads to the increase of the number of analysed impressions/review to more than 2000 texts in which at least one keyword can be found in the reviews' title or content, as it is shown by experiments 3 and 4 from Table 2.3.1. Analysing these results, we observe with high certainty that each of the *PageRank* or *Degree* methods for computing keywords is better than *tf-idf*. However, we note that the cross-checking of the list of obtained keywords with the reviews' title is not effective, as in most cases titles contain words that more or less summarise the reviews, while often the words occurring in the title are not present in the review content. Moreover, we recommend to use only nouns and adjectives as possible keywords, as in our experiments they gave the best results with less computational effort.

Regarding cohesion and word distribution experiments, we were able to observe the small-world phenomenon in the majority of the analysed networks. We support this assumption by observing that the values of the average clustering coefficient were higher than of randomly generated networks with the same number vertices. The values of the *Average Path Length* and *Diameter* metrics were relatively low, compared to the number of vertices, as can be noticed in Figure 2.3.6 that displays the histogram of the obtained values for the *Diameter* metric. We also observed that approximately 64% of the networks had only one connected component, while the rest had 2 or 3 connected components. Moreover, every network contained a Giant Component, containing more than 95% of the nodes. Hence, even if 2 or 3 components were found then one of them was the dominant, while the others were relatively small (counting the number of nodes). For example, Figure 2.3.7 and 2.3.8 show the graphical representation of a network with 2 respectively 3 connected components. The presence of the small-world phenomenon and dominance of one giant component confirms that the reviews were not randomly, but rather naturally generated, as they were composed and

Table 2.3.1: Keyword extraction results. The columns *Title* and *Text* indicate the parts of speech that were kept after filtration: N = noun, A = adjective, Ad = adverb, V = verb and M = location metadata. The following columns represent the number of words from the text used to make the comparison. For example, the column labelled 20% means that only the top 20% of words for the given method of extraction (indicated by the row label) were considered. So, for column 50% and row 5 we should read the result as following: on average, in 14% of texts' titles we could find words from the top 50% words ranked by *PageRank* when from the title we consider only nouns, adjectives and location metadata and from the texts we consider only nouns and adjectives. The *Nr. reviews* column indicates the number of reviews out of the 2542 from our data set on which the statistics were made.

Nr.	Method	Title	Text	100%	50%	33%	Nr. reviews
1	degree	NA	NA	0,1199	0,05168	0,0319	2542
2	degree	NA	NA	0,4335	0,1867	0,1154	703
3	degree	NAAdV	NAAdV	0,3671	0,1606	0,0687	723
4	degree	NAAdVM	NAAdV	0,2715	0,1328	0,0837	2184
5	PageRank	NAM	NA	0,2868	0,1400	0,0887	2171
6	PageRank	NAAdVM	NAAdV	0,2717	0,1326	0,0838	2181
7	PageRank	M	NA	0,3688	0,1791	0,1104	2081
8	Tfi Idf	NAM	NA	0,1736	0,0817	0,0593	2107
Nr.	Method	Title	Text	20%	10%	5%	Nr. reviews
1	degree	NA	NA	0,0233	0,0111	0,0066	2542
2	degree	NA	NA	0,0844	0,0404	0,0240	703
3	degree	NAAdV	NAAdV	0,0989	0,0328	0,0192	723
4	degree	NAAdVM	NAAdV	0,0580	0,0264	0,0139	2184
5	PageRank	NAM	NA	0,0618	0,0271	0,0148	2171
6	PageRank	NAAdVM	NAAdV	0,0579	0,0253	0,0137	2181
7	PageRank	M	NA	0,0748	0,0343	0,0179	2081
8	Tfi Idf	NAM	NA	0,0390	0,0221	0,0129	2107

written by human reviewers. Moreover, we noticed the scale-free distribution of the nodes' *Degree,* (nodes representing tokenized words), an observation falling within the general trend of networks acquired from lexical corpora, as mentioned in [185].

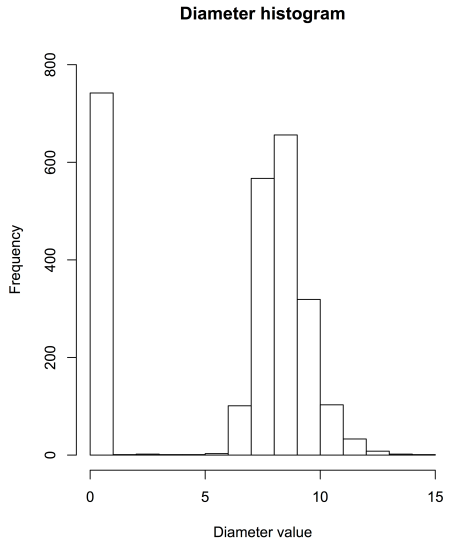


Figure 2.3.6: *AmFostAcolo*'s Histogram of the *Diameter* values obtained for the networks defined by reviews.

2.4 Conclusions & Future Work

In this chapter, we have used several computational methods, techniques and tools stemming from various fields: Complex Network Analysis, Natural Language Processing, Graph Theory, and Graph Databases, to analyze a portal for sharing touristic opinions (*AmFostAcolo*). The analysis was inspired by the portal functionalities, including question asking and answering, impression and review posting, as well as commenting and echoing. The main goal of the study was to achieve a better understanding of the touristic collaboration phenomenon.

Regarding the data extracted and analysed from the question-answering functionality, we draw a number of conclusions. We have proven the usefulness of using the Neo4j graph database model for retrieval of information that is relevant for two real-world touristic scenarios. The network science experiments allowed us to state that the network is stable and resilient, while continuing a cyclical trend of expansion that attracts new users and then matures their connection. The presence of various social phenomena has shown that the social environment has meritocratic traits, i.e. users with more intense activity are better trusted while acting as

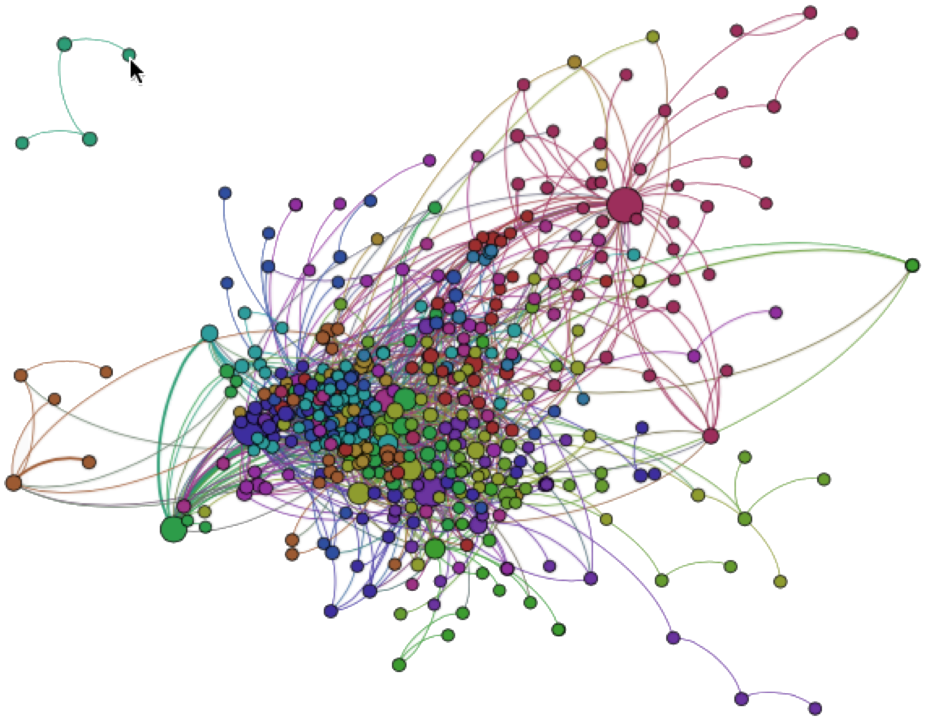


Figure 2.3.7: *AmFostAcolo*'s example of a network with 2 connected components. The radius of each node is proportional to its *PageRank* coefficient. The colour of each node denotes membership to a community, as computed with the *Modularity* algorithm.

enhancers of information diffusion. Moreover, we have proven that information travels fast and easy even if the source user is peripheral. We were able to identify inner-communities of users and give evidence that the collaboration among the communities is intense. We have also experimented and proposed a NLP-based method to identify touristic traits of each community.

Using the texts extracted from the impression-echoing functionality, we proposed a new method, based on NS and NLP that is able to summarise the texts via keyword extraction. The results have shown that our two proposed methods (*Degree* and *PageRank*) perform better than the widely used *tf-idf* information retrieval method. Also, we have brought evidence that nouns and adjectives are the parts of speech carrying out the most relevant touristic information. Nevertheless, the use of location metadata has proven useful for our experiments and we claim

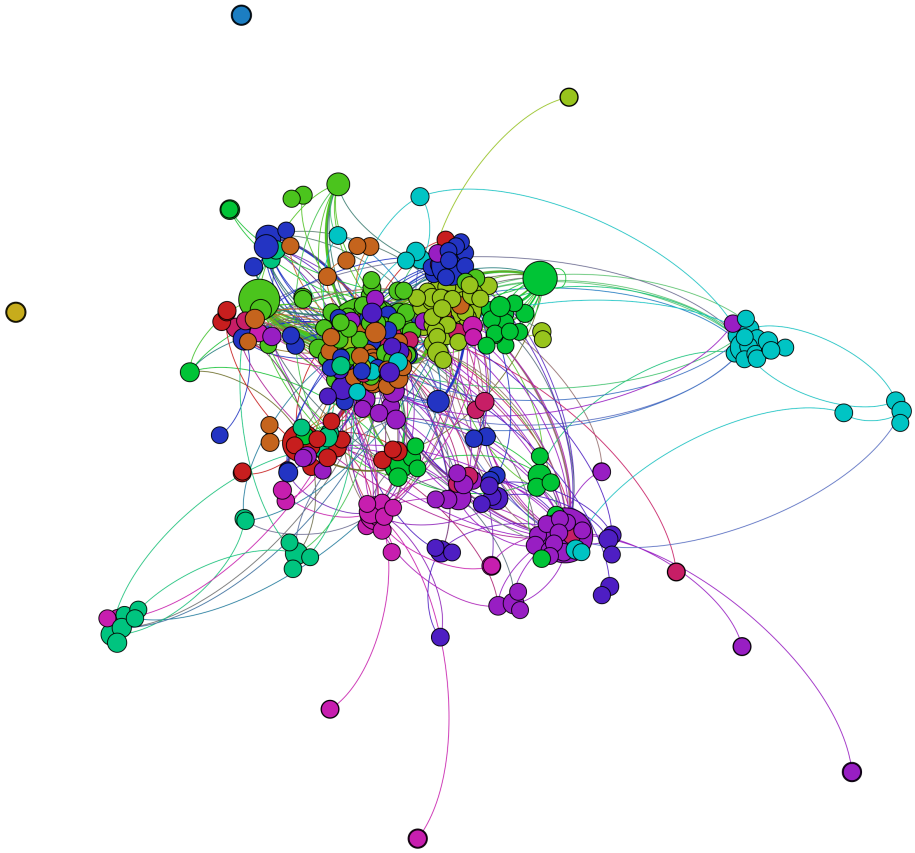


Figure 2.3.8: *AmFostAcolo*'s example of a network with 3 connected components. The radius of each node is proportional to its *PageRank* coefficient. The colour of each node denotes membership to a community, as computed with the *Modularity* algorithm.

that it carries out relevant information. Furthermore, we have applied NS methods to determine the cohesion of the posted reviews and their similarity to real-world texts. This is useful to check that the reviews were not automatically generated by software bots.

As a general conclusion, we have proven that our proposed approaches can bring new valuable insights on how tourists collaborate via touristic portals. However, we acknowledge that this study can be expanded by adding more touristic scenarios and eventually also expanding the data model. A further analysis is required towards inspecting visitation trends, based on previous studies of complex

networks. Regarding our proposed keyword extraction methods, we acknowledge the fact that validating the keywords based on words in the title has its caveats, thus other validation methods need to be developed. However, as current results have shown our proposed method gives better results than *tf-idf*.

I never teach my pupils. I only attempt to provide the conditions in which they can learn.

Albert Einstein

3

Social Learning Environments

According to social learning theory [16], people can learn by observing and modelling the behaviour and attitudes of others. Understanding is socially constructed by means of conversations and interactions around specific problems and activities. Learning is seen as a product of participation in a community, rather than an independent, individual process with social aspects [4]. With the advent of social media, learning by observing the activities, productions and discussions of experts and peers is made possible on a new level [4] [177].

Therefore, in recent years, social media tools have found their way into the educational landscape, being used to foster communication and collaboration between learners, help build communities and encourage positive interactions [69]. Practical usage scenarios for various categories of social media tools are well summarised in [54], [170]. For example, wikis can be used to co-create content, collaboratively edit a document, incrementally accumulate and organise knowledge and integrate resources from different web sources, see [216]. Blogs [213] or micro-blogging services [123] can provide support for writing a group learning

diary, for creating an e-portfolio, for asking and receiving help from peers. Social bookmarking tools [72] can help students discover relevant educational resources, which may be further tagged, organised and shared with peers [177].

So far, few studies have explored the relations between students' behaviour in social media-based learning environments (i.e., patterns of interaction with the tools) and their learning styles [66, 179]. Previous studies have found that: i) active students tend to post more frequently to their blogs than reflective students; ii) reflective students' ratio of reading other blog postings versus posting to their own blogs is significantly higher than that of active students; iii) active students use charts displaying the number of postings and peer rating more often than reflective students; iv) sequential learners tend to write longer posts than global learners [66].

With the expanding popularity of social media tools in Computer Supported Collaborative Learning (CSCL), the need to examine how individual learners interact differently in online communities has increased. In this context, we aim to evaluate how the learners' writing style in social media environments can be used to predict their overall learning performance, as well as their individual learning style. To do so, we use multiple textual complexity indices ranging from lexical, syntactical to semantic analyses [5], described in detail later on in this chapter, to create an in-depth perspective of each learner's writing style. This style analysis creates a basis for predicting the overall performance of each learner in their educational scenario, as well as for highlighting specific traits of their learning style.

However, examining quantitative behaviour indicators is only one approach to assessing learning behaviour and success. The other approach uses Social Network Analysis methods to examine students social behaviour and present it as meaningful knowledge to class instructors. Thus, we shall start by proposing a conceptual knowledge extraction framework for social learning environments based on SNA. Then, we shall conduct a framework directed analysis on real data gathered from a CSCL course, with the purpose of validating and enhancing the proposed framework. Last, but not least we shall introduce and develop a tool for analysing and visualising students' collaborations, orieted towards the average computer literate instructor.

3.1 Background & Related Work

3.1.1 Educational Context

The educational context was a course on "Web Applications Design" (WAD) taught to 4th year undergraduate students in Computer Science. A project-based learning (PBL) approach was used in which learning was organised around the development of web applications comprising of multiple state-of-the-art technologies. Students collaborated in teams of 4 peers in order to build their chosen application (e.g., a virtual bookstore, an online auction website, a professional social network, an online travel agency, etc.). In addition, students had to create four compulsory intermediary presentations in order to be actively engaged throughout the semester and to discourage the practice of activity peaks at the end.

The PBL scenario was implemented in blended mode, with weekly face-to-face meetings between each team and their instructor. These meetings were complemented by the use of three social media tools (wiki, blog and a micro-blogging tool) for online communication and collaboration. MediaWiki[1] was used for collaborative writing tasks, for gathering and organising team knowledge-base and resources, and for documenting the project. Blogger[2] was used for reporting the progress of each project similar to a "learning diary", for publishing ideas and resources, as well as for providing feedback and solutions to peer problems. Each team had its own blog, but inter-team cooperation was encouraged as well. Twitter[3] was meant for encouraging additional connects to peers and for posting short news, announcements, questions, and status updates regarding each project.

These three CSCL tools were all integrated in a social learning environment called *eMUSE* (empowering MashUps for Social E-learning) [178]. The platform provided support for both students and teachers as a unique access point to the social media tools, basic administrative services, learner tracking and data visualisations, as well as evaluation and grading support. *eMUSE* retrieves learner actions from each of the disparate Web 2.0 tools (by means of open APIs or Atom/RSS feeds) and stores them in a local database, together with a description and an as-

[1] http://www.mediawiki.org/
[2] www.blogger.com
[3] https://twitter.com

Table 3.1.1: Distribution of students and contributions per academic year.

	2010-2011	2011-2012	2012-2013	2013-2014	2014 - 2015	2015 - 2016
Nr. of students	45	48	56	66	53	75
Nr. of blog posts & comments	166	121	318	1074	451	479
Nr. of tweets	326	181	1213	1561	956	1223

sociated time-stamp. The yearly distribution of students and of their social media contributions is presented in Table 3.1.1.

3.1.2 Performance Analysis & Prediction

Automated prediction of student performance in technology enhanced learning settings is a popular, yet complex research issue [15, 191]. The popularity comes from the value of the predictive information which can be used for advising the instructor about students at-risk, who are in need of more assistance [158, 234]. More generally, automated methods offer instructors the ability to monitor learning progress and provide personalised feedback and interventions to students in any performance state [232]. In addition, individualised strategies for improving participation may also be suggested [234]. Furthermore, a formative assessment tool could be envisaged based on the automatic prediction mechanism [234], which has the potential to decrease instructors' assessment loads [232]. Finally, students' awareness can be increased by providing them prediction results and personalised feedback [232].

Performance prediction has been extensively studied in web-based educational systems and, in particular, in Learning Management Systems (LMS). This is due to the availability of large amounts of student behavioural data, automatically logged by these systems, such as: visits and session times, accessed resources, assessment results, online activity and involvement in chats and forums, etc. [191]. Thus, student performance prediction models based on *Moodle* log data have been proposed in multiple previous studies [48, 190, 236]. Additionally, log data from intelligent tutoring systems (ITS) have also been used for performance prediction [174]. As far as computational techniques are involved, a wide variety of methods have been applied for predicting students' performance, such as linear regres-

sion [186], neural network models [48], Support Vector Machines and k-nearest neighbours [190], Bayesian networks [174], rank correlation analysis [66], mixed effects analysis of variance (ANOVA) models [123], principal component analysis [97], as well as machine learning algorithms for classification, association rule induction and feature selection [179], or genetic algorithms [232, 236].

In contrast, students' engagement with social media tools in emerging social learning environments has been less investigated as a potential performance predictor [97, 123]. Some preliminary results suggest that the number of blog posts, wiki page revisions, and shared bookmarks are reliable predictors of student success [97].

3.1.3 Textual Complexity Assessment

Textual complexity analysis can be used both for identifying the most appropriate reading material according to students' comprehension level and for assessing students' writing style and knowledge level from their writing traces, which is our current research goal. According to [8], measuring textual complexity can be split into three perspectives: qualitative, reader/task orientation, and quantitative. The qualitative dimensions of textual complexity cover various levels of meaning, structure, language conventionality and knowledge requirements. Reader and task considerations are associated with readers' motivation, knowledge and interest. Quantitative factors are the core of our analysis because they create the basis for automated methods and tools which can be used to analyse textual complexity.

Such automated methods have been developed into frameworks that have demonstrated success in understanding learning behaviours in collaborative learning environments [162]. These frameworks include: *Lexile* (MetaMetrics), *ATOS* (Renaissance Learning, *Degrees of Reading Power: REAP* (Carnegie Mellon University), *SourceRater* (Educational Testing Service), *Coh-Metrix* (University of Memphis), *TAALES* and *TAACO* (Georgia State University). The implemented framework that we use in this study, *ReaderBench* [60, 61], covers a wide range of lexical, syntactic, semantics and discourse centred textual complexity indices, including the most frequent indices from the above-mentioned systems. In addition, *ReaderBench* focuses on text cohesion and discourse connectivity, and provides a more

in-depth perspective of discourse structure based on Cohesion Network Analysis [62]. Cohesion Network Analysis is used to model the semantic links between different text constituents in a multi-layered cohesion graph [214]. The following subsections present the integrated dimensions of analyses possible within *ReaderBench* (RB).

Surface Analysis

Categories like fluency, diction and basic readability formulas relying on surface indices (e.g., words, commas, phrase length, periods) are computed in RB as a way of evaluating lexical and syntactic levels of text difficulty. Page and Wresch [172, 230] have demonstrated that static attributes can effectively predict essay scores. Page's work on quantifying an essay's complexity has led to the identification of correlations between *proxes* (computer approximations of interest) and *human trins* (intrinsic variables – human measures used for evaluation). Starting from Page's metrics and taking into consideration Slotnick's categories [203] grouping proxes based on their intrinsic values, multiple indices from their studies have been integrated within our model including: average paragraph/ sentence/-word lengths in characters, average and standard deviation of paragraph/sentence lengths in words (including separate indices for unique content words), as well as number of commas per sentence/paragraph. In addition, entropy [199], which can be defined as the expected value of the information contained in the text, is used in RB to evaluate lexical diversity of word stems and characters within the input text.

Word Complexity

In the first step of the RB processing, words are extracted from the input text. A Natural Language Processing (NLP) pipeline then splits the words, tokenizes them, eliminates stop words, and then conducts Part Of Speech (POS) tagging, lemmatization, parsing and Named Entity Recognition (NER) [145]. RB calculates a number of indices for content words in the text. These indices include *syllable count, distance between the inflected form, lemma and stem, corpus frequency, distance within the hypernym tree and word polysemy*. These indices are used to ap-

proximate each word's individual complexity. *Corpus frequency* is computed as an inverse frequency of words from the training corpora, while *polysemy* is computed on the basis of the lexicalized ontology WordNet [156]. In general, words with multiple senses have a higher complexity because it is harder to assign the correct sense to them. The *distance within the hypernym tree* is related to a word's specificity and is determined as the distance to the ontology's root; longer paths usually indicate specialisation or specificity for given words. In addition, the differences between the *inflected form*, the *lemma* and the *stem* reveal the use of multiple juxtaposed prefixes and suffixes, which is another mark of a word with a higher complexity.

Syntax

The most predictive POS tags in terms of textual complexity are prepositions, adjectives and adverbs [162] that allow for a more detailed and complex text structure. In addition, RB uses a syntactic parser to calculate a number of syntactic indices (e.g., *overall size of the parsing tree, maximum depth, number of semantic dependencies*). Higher values for these indices usually indicate greater complexity [96].

Semantics

According to McNamara et al.[149], textual complexity is also linked to cohesion with regard to text comprehension. Cohesion is a central element for obtaining a coherent mental representation of discourse, commonly called a situatio model [221]. Therefore, texts that lack cohesion may be perceived as having a higher difficulty due to an increased cognitive load on the part of the reader. In RB, a cohesion graph is used to model the underlying structure of discourse [214]. Cohesion is determined at both inter- and intra- paragraph levels of analysis based on three semantic distances: a) Wu-Palmer semantic similarity in WordNet [231], b) cosine similarity using Latent Semantic Analysis vector spaces [133] and c) Jensen-Shannon dissimilarity between Latent Dirichlet topic distributions [31]. In general, a text is more complex as the number of relevant links in the cohesion graph increases. Thus, the average value of all the inter- and intra- paragraph

links reflects text complexity [214]. In addition, specific discourse connectors defined as cue phrases are also considered in order to evaluate the degree of discourse elaboration.

Named entity derived features also influence comprehension, because they create the basic components of concepts and propositions on which higher-level discourse processing is based [78]. Therefore indexes based on named entity statistics are also included in the RB framework.

Social Network Analysis

Since *eMUSE* supports the creation of learning networks, which can be modelled as social graphs, a knowledge extraction process can be performed by applying Social Network Analysis (SNA) methods [20, 196] .

SNA has also been used in technology-enhanced learning area, as summarised in [189]. The paper focuses especially on the use of collaborative filtering techniques, which can generate personalised recommendations for students; learning objects, relevant links, relevant courses or most appropriate study partners can be the subject of these recommendations. Various communication aspects, patterns of academic collaboration and the structure of online learning communities have also been investigated, according to [189].

More recently, Maglajlic and Gutl [143] also used SNA in an educational environment in order to observe, measure, enhance collaboration and as an early detection method of potential weak trainees. Also, they proposed a social engineering algorithm with the purpose of placing a trainee in the appropriate tutored group. The authors employed SNA techniques such as cliques, centrality and density. Furthermore, the authors brought evidence that the efficiency of digital learning environments can be monitored through SNA.

Crespo and Antunes [55] focused their attention on the representation and analysis of teamwork in an educational context. They found that SNA can be used with success in representing, exploring and predicting teamwork results. As SNA techniques, the authors used diverse variants of the *PageRank* algorithm for ranking learners.

Haythornthwaite [106] also investigated social networks analysis methods and

metrics for e-learning, based on specific educational needs. The following SNA methods were selected for use: number of ties, density, centrality and cliques. The authors further discussed the perspectives of SNA for understanding social learning in [107]. In addition to an overview on the use of SNA techniques in learning networks, the paper includes a discussion on how to create/interpret a social graph. In designing our framework, we tried to answer some of the questions raised by the authors, e.g. What metrics matter for networked learning? [107].

As for software tools used for SNA analysis we mention those that we will use extensively throughout this chapter, while others will be mentioned where needed. *Gephi* [4] is a visualisation and exploration software for graphs. *Pajek* [5] is a software tool for large graph analysis; although it lacks *Gephi*'s ease of use, it comprises many more SNA methods. *NetworkX* [6] [105] and *GraphViz*[7] represent mature programming libraries oriented on the study of graphs.

Information Visualisation

Information visualisation relies on the remarkable visual perception abilities of humans for pattern discovery [233]. It employs interactive visual representations in order to amplify cognition [223] and generate *insight* [46].

Visual approaches have been used in learning analytics, to help teachers and students explore learner traces from virtual learning environments. Various types of data can be included in a learning analytics dashboard, such as: artefacts produced by learners, social interaction, resource use, time spent, test and self-assessment results. The goal is to provide insight into learning data, supporting awareness and decision making, and increasing students' engagement and motivation [223].

Networks or graphs are a common visualisation method in educational settings [46]. They can be used to display information regarding students' interactions, which is particularly important in case of collaborative learning and social learning environments.

In particular, social network analysis (SNA) and network visualisations have

[4] https://gephi.org/
[5] http://vlado.fmf.uni-lj.si/pub/networks/Pajek/
[6] https://networkx.github.io/
[7] https://graphviz.org/

been used to investigate students' interactions taking place in educational environments [55, 189].

3.2 Predicting Students' Performance and Differences

The approach that we propose in this section is based on the content analysis of each student's contributions as opposed to student behaviours. Specifically, in this study, we apply textual complexity analyses on blog entries and tweets posted by students in the context of *eMUSE* collaborative project-based learning scenario. We also use survey results to assess students' learning styles. We use the textual analysis to predict learning outcomes and learning styles.

In the following we introduce two sets of experiments. The first one analyses a broader range of data in order to obtain a general perspective. The second experiments reduces the range of the data used for a more in depth perspective.

3.2.1 Data Acquisition & Preprocessing

Data Acquisition

For the first experiment, data was collected over 6 consecutive winter semesters (2010/2011 – 2015/2016), from the WAD course. Small improvements and refinements were made from one year to the consecutive one, based on students' feedback and instructor experience. A total of 343 students, enrolled in the WAD course, participated in this experiment. For details on the number of contributions see section 3.1.1.

For the second experiment, the data was collected during the first semester of the 2013-2014 academic during the WAD course. The participants in the second experiment consisted of 66 undergraduate students split into 17 teams (16 with 4 members and 1 with 2 members). For the current experiment, the writing actions we used to assess student writing style were students' tweets, together with their blog posts and comments. The total number of student contributions recorded at the end of the semester included 1561 tweets, 708 blog posts and 366 blog comments.

According to Keefe [126], learning styles refer to the individual manner in

which a person approaches a learning task, as well as their preferences related to perception modality, processing and organising information, reasoning or social aspects. Various learning style models have been proposed during the past decades, which differ in terms of the underlying learning theories, as well as the number and descriptions of included dimensions. One of the most popular models in technology-enhanced learning is FSLSM proposed by Felder and Silverman [66, 77]. According to FSLSM, learners are categorised in terms of their preferences based on four dimensions: a) active versus reflective; b) sensing versus intuitive; c) visual versus verbal and d) sequential versus global [77].

To collect students' learning styles according to FSLSM [77] we used a dedicated inventory: the Index of Learning Styles questionnaire (ILS) [205]. The ILS consisted of 44 questions, each with two possible answers. As a result of the test, the learning style of the student was described on a scale between -11 and $+11$ (with a step of $+/-2$) for each FSLSM dimension. As an example, a score of $+11$ on the active/reflective dimension implies a strong active preference, while a score of -3 implies a mild reflective preference. The ILS questionnaire was not mandatory, and, as a result, only 48 students completed it.

Data Preprocessing

Data preprocessing was necessary because many texts contained elements that added no value to our research (e.g., HTML tags, images, emoticons, references to other students, computer code, URLs). Also, students used both English (second language) and Romanian (native) languages in their blog and Twitter contributions. However, textual complexity cannot be performed in a cross-language manner due to each language's specificity. Thus, we had to separate English from Romanian texts. For the purpose of this study, only English contributions were considered.

The first preprocessing step focusing on cleaning the input texts was automatically performed by means of an extension of the BeautifulSoup[8] library. After applying the previous cleaning mechanisms, we performed language detection based on the langdetect[9] library (ported from Google's language detection software). Next we filtered in only the students who had at least five English contributions

[8] https://pypi.python.org/pypi/beautifulsoup4
[9] https://pypi.python.org/pypi/langdetect

after preprocessing. Also, a threshold of minimum 50 content words was considered for our textual complexity analysis. We defined a content word as dictionary word, we do not consider stop-words (common words with little meaning - e.g., "and", "the", "an"), which has as corresponding part-of-speech a noun, verb, adjective or adverb.

Thus, for the first experiment a total of 148 students were included in our analysis, having cumulatively 3013 textual contributions. Next we modelled the data as one week time-frames (i.e., 7 days starting from each consecutive Monday) due to the schedule of the academic semester in which students had one WAD class per week. The total length of the considered time series is 16 weeks, including 14 weeks of classes and 2 weeks for the winter holidays. For each student, the number of weekly blog and micro-blog posts was computed in order to obtain his/her time series of social media contributions.

As the amount of data for the second experiment was less than in the first experiment, manual language corrections were done (i.e., spelling). Besides spell-checking, we also enriched the texts with expanded Twitter tags, if present. For example, if a semantically relevant twitter tag was found (e.g., #workinghard), then the words contained within the tag were extracted and added to the text. Thus, our statistical analysis for the second experiment was performed on 29 students having 848 textual contributions, out of which 27 finished the course and only 25 had completed the ILS questionnaire.

3.2.2 Results & Discussion

First Experiment

The performed longitudinal analysis relies on a wide range of evolution indices including *average & standard deviation of contributions, entropy, uniformity, local extreme points*, and *average & standard deviation of recurrence*. We refer readers to [5] for further information about these features that were initially used for keystroke analysis.

We split the students into two equitable groups: high performance students with grades 8, while the rest were catalogued as low performance students. The indices from *ReaderBench* and from the longitudinal analysis that lacked normal

Table 3.2.1: Correlations between *ReaderBench* and longitudinal analysis indices, and course grade.

Index	r	p
Word entropy	.416	<.001
Time series entropy	.378	<.001
Average verbs per sentence	.323	<.001
Avg. cohesion (LDA) between sentences and corresponding contribution	-.274	<.010
Avg. unique words per sentence	.270	<.001
Avg. prepositions per sentence	.264	<.010
Time series local extremes	.236	<.010
Avg. adverbs per sentence	.236	<.010
Avg. pronouns per sentence	.250	<.010
Avg. named entities per sentence	.189	<.050

distributions were discarded. Correlations were then calculated for the remaining indices to determine whether there was a statistical ($p < .05$) and meaningful relation (at least a small effect size, $r > .1$) between the selected indices and the dependent variable (the students' final score in the course). Indices that were highly collinear ($r.900$) were flagged, and the index with the strongest correlation with course grade was retained, while the other indices were removed. The remaining indices were included as predictor variables in a stepwise multiple regression to explain the variance in the students' final scores in the WAD course, as well as predictors in a Discriminant Function Analysis [15] used to classify students based on their performance.

Medium to weak effects were found for *ReaderBench* indices related to word entropy, number of verbs, prepositions, adverbs, and pronouns, the number of unique words, number of named entities per sentence, and average cohesion between sentences and corresponding contributions measured with *Latent Dirichlet Allocation* [60] (see Table 3.2.1).

We conducted a stepwise regression analysis using the ten significant indices as the independent variables. This yielded a significant model, $F(3, 143) = 17.893$, $p < .001$, $r = .521$, $R2 = .272$. Three variables were significant and positive predictors of course grades: word entropy, time series entropy and average verbs in sentence, denoting a higher activity and participation for high performance students. These variables explained 27% of the variance in the students' final scores for the course.

Table 3.2.2: Confusion matrix for DFA classifying students based on performance.

		Predicted Performance Membership		Total
		Low	High	
Whole set	Low	48	23	71
Whole set	High	17	60	77
Cross-validated	Low	48	23	71
Cross-validated	High	20	57	77

The stepwise Discriminant Function Analysis (DFA) retained the same three variables as significant predictors of course performance (Time series entropy had the highest standardized canonical discriminant function coefficient), and removed the remaining variables as non-significant predictors. These three indices correctly allocated 108 of the 148 students from the filtered dataset, $\chi^2(df = 3, n = 148) = 43.543 p < .001$, for an accuracy of 73.0% (the chance level for this analysis is 50%). For the leave-one-out cross-validation (LOOCV), the discriminant analysis allocated 105 of the 148 students for an accuracy of 70.9% (see the confusion matrix reported in Table 3.2.2 for results). The measure of agreement between the actual student performance and that assigned by the model produced a weighted Cohen's Kappa of .457, demonstrating moderate agreement.

Second Experiment

Due to the limited amount of data and the need to control for over-fitting, *2-tailed Pearson correlations* were first computed in order to determine which textual complexity indices were highly correlated to the project and exam grades (see Table 3.2.3).

Two stepwise regression analyses were performed in order to determine the degree to which the three automated indices with the highest correlations predicted students' project and exam grades. Both regressions yielded significant models, $F_{projectgrade}(1, 25) = 30.981, p < .01, r = .744, R2 = .553$; $F_{examgrade}(1, 25) = 12.331, p < .01, r = .575, R2 = .330$. For each regression, only one variable was a significant predictor (average named entities per contribution). This index accounted for 55% of the variance in the project grading, [$\beta = .744, t(1, 25) = 5.46,$

Table 3.2.3: Correlations between textual complexity indices and project/ exam grades (N = 27)

	Project Grade	p	Exam Grade	p
Avg. named entities per contribution	.744**	0	.575**	.002
Avg. commas per contribution	.517**	.006	.371	.057
Avg. unique words per contribution	.502**	.008	.413*	.032
Avg. indefinite pronouns per contribution	.493**	.009	.430*	.025
Avg. first person pronouns per contribution	.488**	.01	.434*	.024
Avg. reason and purpose connectives per contribution	.482*	0.11	.355	.069
Avg. sentence - contribution cohesion via LDA	-.481*	.011	-.423*	.028
Avg. sentences per contribution	.478*	.012	.481*	.011
Avg. temporal connectors per contribution	.464*	.015	.377	.052
Avg. coordinating connectives per contribution	.376	.054	.325	.098
Word entropy	.363	.063	.289	.143
Avg. unique words per sentence	.335	.088	.179	.371
Avg. simple subordinates per contribution	.333	.089	.289	.144

$p < .01$] and 33% of the exam grade.

Afterwards, a stepwise Discriminant Function Analysis (DFA) was used to classify students on each learning style dimension. For this analysis, +/-1 and +/-3 values (as resulted from the ILS questionnaire) were considered neutral, whereas the other values per dimension were catalogued as positive/negative. As with the regression analysis, only the top 3 indices with the highest effect size were considered in order to control for over-fitting.

Regarding the *Active/Reflective Dimension*, the stepwise DFA retained two variables and removed the remaining variables as non-significant predictors. The results demonstrate that the DFA using these two indices correctly allocated 15 of the 25 texts in the total set for an accuracy of 60% (the chance level for this analysis is 33%). See Table 3.2.4 for variables and results. The measure of agreement between the actual text type and that assigned by the model produced a weighted Cohen's Kappa of 0.256, demonstrating fair agreement

For the *Sensing/Intuitive Dimension*, the stepwise DFA indicated that no variables were predictive of the sensing – intuitive dimension.

On the *Visual/Verbal Dimension*, the stepwise DFA retained three variables and removed the remaining variables as non-significant predictors. The results demonstrate that the DFA using these three indices correctly allocated 20 of the 25 texts in the total set for an accuracy of 80% (the chance level for this analysis is 33%). See Table 3.2.4 for variables and results. The measure of agreement be-

Table 3.2.4: Discriminant Function Analysis results (N = 25)

FSLSM dimension	Significant predictors	χ^2	p	Accuracy
active - reflective	Avg. conjunct connectives per contribution Average conditional connectives per contribution	25.781 (df = 4)	.001	60%
sensing - intuitive	No predictive model could be trained			
visual - verbal	Avg. conjunct connectives per contribution Average conditional connectives per contribution Avg. disjunction connectives per contribution	32.469 (df = 4)	.001	80%
sequential - global	Avg. reason and purpose connectives per contribution	10.164 (df = 2)	.038	48 %

tween the actual text type and that assigned by the model produced a weighted Cohen's Kappa of 0.583, demonstrating moderate agreement.

Regarding the *Sequential/Global Dimension*, the stepwise DFA retained one variable and removed the remaining variables as non-significant predictors. The results demonstrate that the DFA using this index correctly allocated 12 of the 25 texts in the total set for an accuracy of 48% (the chance level for this analysis is 33%). See Table 3.2.4 for variables and results. The measure of agreement between the actual text type and that assigned by the model produced a weighted Cohen's Kappa of 0.188, demonstrating fair agreement.

3.3 Conceptual Knowledge Extraction Framework

Investigating students' interactions and the collaborative learning processes taking place in these social learning environments is an important research direction, which could bring valuable insight into the collaborative learning process. Hence, what we propose in this section is a conceptual framework for knowledge extraction and visualisation from a social media-based learning environment. In particular, we will focus on *eMUSE* platform, proposed in [178]; more details about

Table 3.3.1: Extracted set of Knowledge Needs

ID	Short Description
KN1	Determine the collaboration network.
KN2	Determine methods to quantify collaboration over various time intervals and various network granularities.
KN3	Determine salient students or communities of students.
KN4	Determine if the course environment adequately supports collaboration.
KN5	Determine external factors that influence the evolution of collaboration.
KN6	Determine the impact of instructors' actions on collaboration.
KN7	Determine clear and comprehensive methods of presenting the information extracted.

eMUSE are provided in section 3.1.1. The declared objective of the proposed framework is to identify appropriate SNA techniques and integrate them in a conceptual knowledge extraction framework.

3.3.1 Basis & Rationale

Our framework proposal builds on the previous papers, see section 3.1.3, and aims to cover SNA methods that have the potential to answer specific instructional needs and have been proven useful in similar scenarios. The proposed conceptual knowledge extraction framework is designed on the basis of *eMUSE* [178] social learning environment. Hence, we will select only SNA methods that could be used on the data provided by *eMUSE*. Furthermore, the analysis methods have to address certain knowledge needs defined by instructors working with *eMUSE*.

Knowledge Needs

The general purpose of our framework is to provide useful knowledge from the instructors' point of view. Hence, we asked the teachers working with *eMUSE* to define their knowledge needs (KN). Next, we filtered the KN by keeping only those that we could partially/totally quantify and satisfy through the use of SNA methods/techniques. Then, a series of refinement and clustering stages were conducted through the instructors' continuous feedback. The compiled list of extracted knowledge needs is included in Table 3.3.1.

KN 1 is an obvious need as there cannot be any analysis without building the collaboration network. This need will be treated in the next subsection.

KN 2 implies the use of current SNA methods and possible development/adaptation of new techniques to capture collaboration. The development/adaptation of new methods is justified by their use in a new context, the digital learning context.

Salient students or communities are those that are usually at the high end of the collaboration spectrum. By satisfying KN 3 instructors should be able to pinpoint students / communities that mostly support collaboration. Also, instructors could be presented with early warnings for students / communities that do not engage in collaboration, those with possible low learning outcomes.

By course environment we refer both to the platform used in the course (in our case, *eMUSE* in conjunction with several social media tools), and to the pedagogical scenario set by the course instructor (in our case, group project-based learning). The environment is an important factor in the evolution of any collaboration network. Hence, knowing if the course environment adequately supports collaboration (KN 4) is a valuable affordance.

KN 5 addresses the discovery of external factors that influence the evolution of the collaboration network. We define external factors as those which are not controlled by the instructor, e.g. social phenomena or students' habits [215], [157]. Having an insight on the presence of such phenomena/habits would give the instructor additional leverage. For example, the presence on preferential attachment phenomenon could explain a high rate of collaboration among some learners.

By satisfying KN 6, teachers would get the means of evaluating their actions based on the impact on the collaboration network.

Instructors should be presented with extracted knowledge in a clear manner, that is easy to understand (KN 7). Thus, the framework adoption effort would be reduced and the focus would shift to the benefits it provides.

Constructing the Social Graph

As already mentioned, the objects of study of our framework are social graphs. A social graph is a graph where persons are depicted through vertices/nodes. Relations among vertices are an abstraction of an interaction manner. In our context, vertices represent learners and interactions represent messages exchanged through

the social media tools integrated in *eMUSE*.

Graph theory defines two manners of representing relations among vertices: oriented edges (arcs) and non-oriented edges (links). We argue that human communication usually implies bidirectional relations; thus, the use of arcs may lead to an oversimplification, as opposed to the use of links. However, by using arcs we can extract knowledge on leadership and pro-activity. Hence, we are advocating the use of both methods for representing social relations, as we can capture diverse facets of inter-human communication.

Since *eMUSE* provides a data collection mechanism which records student interactions on social media, we can easily develop software tools to extract social graphs. We can enrich these social graphs by incorporating the time of message exchanges as an attribute of the arcs/links. Thus, we will be able to develop time-series analyses as requested by KN 2. Furthermore, we can add weights to the arcs/links in order to capture the number of messages exchanged between two students. In addition, the weight can be coupled with the length of the messages (number of characters) or other measures, to better capture the quantitative aspect of communication. Also, other qualitative measures can be used as weighting methods, which we need to identify.

We aim to create as many relevant social graphs as possible, from various facets of communication. We define a facet as a unique way of communication based on the type, digital tool, purpose, etc. Also, graphs can be obtained through aggregation of various other graphs. In our case, we could aggregate the social graphs for each social media tool present in *eMUSE*, thus obtaining a social graph capturing all communication established through the platform.

We consider that the above discussion fully addresses KN 1 (Determine the collaboration network) for our scenario; hence, in what follows we will only focus on the other KNs.

3.3.2 Conceptual Framework Design

Based on the knowledge needs defined in section 3.3.1 and on our experience in using SNA, we identified some methods of knowledge extraction. We consider these methods adequate for our analysis scenario, with a high probability of being

Table 3.3.2: SNA methods for the graph as a whole perspective

Method	KN addressed
Graph Connectivity: No. of connected components, Line connectivity	KN 2, KN 4, KN 6
Information Diffusion: Diameter, Avg. shortest path	KN 2, KN 4, KN 6
Graph Clustering: Global avg. clustering coefficient, No. of K-cliques	KN 2, KN 4, KN 6
Graph ability to support information exchange	KN 4, KN 6
Models of network growth	KN5
Visualisation: Force directed (Force Atlas2, Fruchterman-Reingold), Bow-tie structure	KN 7

useful in similar scenarios; however, we do not imply that the selection is exhaustive. For each method, we will mention the knowledge need(s) that it is able to partially/totally satisfy and give short examples of their respective affordances for an instructor.

The proposed conceptual framework is structured to address three levels of graph granularity. The first level investigates the social graph as a whole. The second level addresses communities that appear inside the graph. Individual vertices in the graph are studied through the framework's third level. Hence, we will be able to explain general traits through individual/group particularities and vice versa. Last but not least, time series analysis perspective will be discussed. As we cannot possibly present each SNA method in detail, we refer the reader to fundamental works such as [20, 34, 58, 119].

Whole Graph Perspective

From the perspective of the whole graph, we selected the set of SNA methods included in Table 3.3.2.

In our educational context, *graph connectivity refers* to the paths of communication between learners. The *number of connected components* is inversely correlated with the ability of the graph to support communication between learners. A *connected component* can be considered a communication silo; no communication is possible with members that are not part of the component. Thus, the *number of connected components* could be seen as a marker for collaboration. In the case

of only one *strong connected component*, we can use line connectivity as another marker for collaboration. Line connectivity is defined as the minimum number of edges whose deletion from a graph would lead to an increase in the *number of connected components*. The higher the value of line connectivity the more communication paths exist; hence, the higher probability of successful collaboration.

In addition, we define *information diffusion* as the number of steps it takes for information to spread out in the entire graph, if it is passed from one learner to another. Hence, in our graph representation of *information diffusion* is related with paths in the graph. The *diameter* of a graph is the longest shortest path between any pair of vertices. The *average shortest path* is computed as the average value of all shortest paths in the graph. The lower the value of the above mentioned metrics, the more direct communication there is. Hence, increasing the chances of successful collaborations. Therefore, *information diffusion* methods are yet another way to capture and evaluate collaboration.

In the real world, people tend to bond together and communities of people arise. Hence, it is obvious that the more bonding there is the more paths of communication emerge. The *global clustering coefficient* represents an overall graph measure of the degree to which vertices in a graph tend to cluster together. Another measure to capture the clustering effect is determining the number of *K-cliques*. A clique represents a complete sub-graph of K vertices. Since in our PBL scenario learners using *eMUSE* generally form teams of 4 members, we would expect the *number of 4-cliques* to be at least equal to the number of teams. Otherwise, it would mean that there are teams where not all members communicate directly, leading to potentially low collaboration and poor course outcomes.

The ability of a social graph to *support knowledge / information exchange* is essential in our scenario, as it affects learners' capability to receive and transmit knowledge. An instructor possessing such an insight can act to modify the social graph to better support knowledge exchange, e.g. encourage active learners to share knowledge among each other. More details on how to determine a graph's ability to support information exchange can be found in [22].

In addition to the traits which can be influenced by the instructor, the graph could exhibit also other traits that appear due to social phenomena or peculiar habits present among the community of learners. Determining the evolutionary

Table 3.3.3: SNA methods for the communities perspective

Method	KN addressed
Community detection: Graph clustering, Modularity	KN 1-7
Important parts discovery: Strong component detection, Island detection, K-cores detection	KN 1-7
Transformation & Visualisation: Reduction, Context, Dendrogram	KN 1-7

mechanism of the network through the *model of network growth* [164], could lead to identifying external factors that impact the collaboration network.

Finally, in order to satisfy knowledge need 7, we will introduce graph drawing methods in our framework. We consider graph visualisations as an easy way to evaluate communication. *Force-directed drawing* algorithms (e.g., *Force Atlas 2* [119] and *Fruchterman-Reingold* [88]) present graphs in an aesthetically pleasant way, with no additional requirements from the instructor, e.g.: as few crossings of links as possible; positioning of learners in a gradient way, with most active learners in the centre; learners belonging to the same community plotted in close vicinity etc

Communities Perspective

We use the term *community* for a group of learners that communicate/collaborate together significantly more than with other learners. From the perspective of the learning communities that appear inside a graph, we selected the set of SNA methods included in Table 3.3.3.

A number of methods can be used for *community detection*. Graph clustering [132] is the task of grouping the vertices of a graph into communities taking into consideration the arcs/links. For an easy visualization of the arrangement of clusters we can use *dendrograms*. Modularity [131] measures the strength and division of the network in communities. Hence, it can be used as a method to determine communities.

Important parts discovery refers to identifying subgraphs with peculiar attributes for further analysis. A *strongly connected component* of a directed graph is a maximal subgraph where there is a path between all vertices. *Island detection* represents

a method to identify relevant parts of a network usually by inspecting the distribution of vertices or arcs'/links' values. A *K-core* [131] is a maximal subgraph with the property that all vertices in the subgraph have a degree equal or higher than K. Thus, we consider *K-core* detection as a method of discovering important parts.

Having identified the learning communities, we can further analyse each one. Considering that a part or a community is a subgraph, which in turn is a graph, we can analyse it with all the methods present in our proposed framework; thus all knowledge needs can be covered.

A *reduction* of a graph represents the merging of constituent parts (e.g. communities) into a vertex for each part (Vpart). Links/arcs between initial vertices of the same constituent part are suppressed. Initial links/arcs between vertices of different constituent parts become links/arcs between their respective Vparts. A *context* transformation is similar to a reduction, with the exception that one initial part is preserved as it is. A graphical illustration of the above mentioned transformations for our learning scenario can be seen in Figure 3.3.1.

Visualising communities/parts and their respective bridges of communication exchange represents a quick and intuitive manner of informing instructors on the general knowledge exchange status. Hence, teachers can easily identify communities that require additional attention. For example, *reduction transformation* could be used to evaluate inter-team communication. In our scenario, this type of communication is neither restricted nor endorsed by the instructors; thus, it is of interest to see if learners break the team boundary and engage in communication with other fellow learners. *Context transformation* could be further used to pinpoint the students, from a specific team, who are engaging in inter-team communication (e.g., student E from Figure 3.3.1).

Individual Perspective

From the perspective of the individual learners inside the graph, we selected the set of SNA methods included in Table 3.3.4.

Centrality measures quantify vertices' importance from a graph theory perspective [58]. In our scenario, these methods try to asses each learner's collaboration

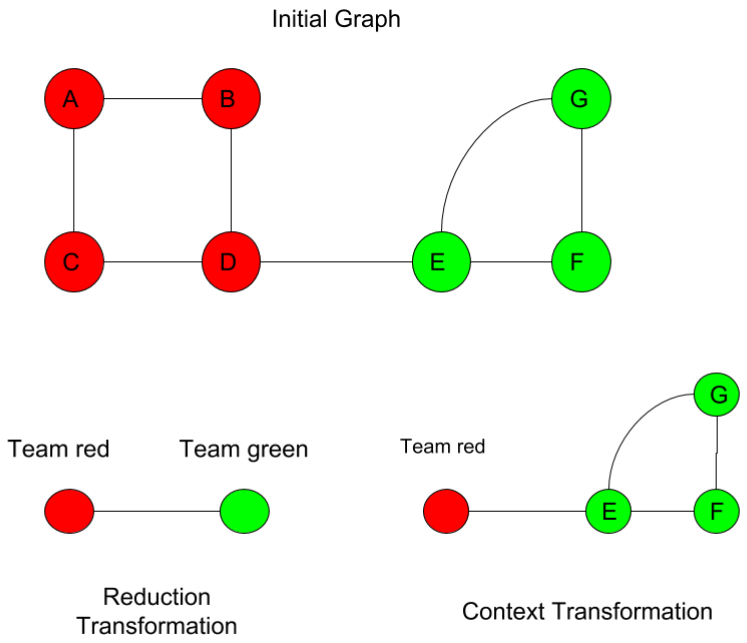


Figure 3.3.1: Graph transformation example; the colour of each vertex defines team membership: students A-D belong to Team "red" and students E-G belong to Team "green".

activity. One of the most popular centrality measures is the *degree* of the vertex. This metric could be used to easily determine the learners that are the most/least active. However, this is a quantitative measure, which gives no indication regarding the quality of the interactions between the students. Hence, a more suitable centrality measure could be *PageRank* [173], *Eigenvector* [35] or *Katz* [125], as they take into consideration qualitative aspects of communication. Salient learners (i.e., those with particularly high or low activity), could be determined by ordering students based on the above centrality measures.

Through the information cycle methods, we are trying to assess which learners are the sources of information (*Hubs*) and which learners are the sinks for information (*Authorities*) [127].

In a graph, a vertex's positioning is of great importance as it may be a critical point in maintaining certain graph traits, e.g. *connectivity*. Hence, learners represented through such vertices are salient. Through *Betweenness* an instructor can

Table 3.3.4: SNA methods for the communities perspective

Method	KN addressed
Measures of centrality: Degree, PageRank, Katz, Eigenvector	KN 2, KN 3, KN 6
Information cycle: Hubs, Authorities	KN 2, KN 3
Positioning: Closeness coefficient, Local clustering coefficient, Betweenness	KN 2, KN 3
Visualisation: Vertez/link scaling, colouring & shaping, Bow-tie structure	KN 7

determine the learners that bridge community silos, those learners that facilitate the exchange of knowledge between communities. The *Local clustering coefficient* is similar with the *Global clustering coefficient* (introduced in 3.3.2), with the difference that it measures a vertex's tendency to communicate with others. The *Closeness* metric is computed as the sum of the lengths of the shortest paths between a vertex and all other vertices in the graph. Hence, learners with low *Closeness* values are positioned on various communication paths, playing an important role in knowledge diffusion.

By using the above mentioned methods, instructors should be able to determine the status of each learner involvement and their recognition by other learners. These insights can be included in the grading methodology, harnessing and encouraging valuable learners.

A *bow-tie* [40] graphical representation of the social graph is able to categorise learners in four categories: sharing information, sharing and acquiring, only acquiring and inactive students. Such a method provides useful information in an easy to understand manner. Through *scaling, colouring & shaping* of links and vertices we can present various attributes in graph drawings, hence making it easy for instructors to visualise multiple facets of communication in a graph representation.

Time-Series Perspective

The *time series perspective* is a valuable complementary method for all the three graph granularity perspectives. It represents a recurrent application of the above mentioned SNA methods at certain points in time or for certain time-frames. Hence,

the evolution of the social network can be visualised throughout time. In addition, the instructor can investigate inflection points and correlate them with learning activities; thus, important affordances can be extracted and harnessed. The time series perspective was introduced to satisfy KN 2.

3.4 Investigating Students' Collaboration Patterns in eMUSE

In the previous section we have introduced a conceptual knowledge extraction framework to explore the collaborative learning process. For the current section we plan to experimentally validate the proposed framework by exploring the collaborative learning process supported by *eMUSE* environment and its integrated social media tools. More specifically, we will investigate students' collaboration patterns in a project-based learning scenario and use SNA to extract knowledge that answers instructors' pedagogical needs, see Table 3.3.1.

3.4.1 Collaboration Overview: Building the Social Graph

Data Source

For this study, we took into account the 2016-2017 instalment of a course on "Web Applications Design", taught to 4th year undergraduate students in Computer Science in the first semester of the academic year, for more details on the course refer to section 3.1.1

A total of 32 students participated in the study, being split in 8 teams (4 students per team). The number of student actions on the social media tools, recorded by *eMUSE* at the end of the semester, included 1686 tweets, 271 blog entries (159 posts and 112 comments) and 1696 wiki page revisions and file uploads.

In the current analysis, we are interested in students' communication and collaboration patterns; since the wiki actions log did not include explicit information on students' interactions, we excluded them from this analysis. We therefore take into account students' communication on the Blog and Twitter, as described next.

Building the Social Graph

A custom tool was designed and implemented for processing the raw data collected by *eMUSE*; Python 3.5 programming language with *NetworkX* graph analysis library were employed for building the social graph. This tool was designed to respect the principles enunciated in section 3.3.1 of the proposed conceptual framework.

More specifically, a directed graph was built starting from students' interactions on the blog and micro-blogging tool; vertices represent learners and links represent messages exchanged through the two social media tools integrated in *eMUSE*. The types of interactions (collaborations) taken into account on Blogger and Twitter respectively are detailed next.

Regarding the Blog, since each team has one common blog space, we assume that each post is addressed to the corresponding team members. Hence, for each blog post we consider a collaboration between the author of the post (source student) and each respective team peer (target students). Thus, 477 collaborations (i.e., $159 posts * 3$ team members) are considered. In addition, each blog comment is directly addressed to the initial blog post author. Hence, a collaboration between the author of the comment (source student) and the author of the initial post (target student) is considered. Thus, for each blog comment we can extract one interaction, resulting in 112 additional collaborations.

As far as Twitter is concerned, collaboration among students is associated to the reference ("@username") and re-tweet mechanisms. This was encountered in 1043 tweets, i.e., 62% of all Twitter actions. For every action presenting the referencing mechanism, we considered a collaboration between the author of the tweet (source student) and all the referenced students in it (target students). Thus, a number of 1635 interactions were extracted from Twitter actions.

Overall, a total of 2224 collaborations were extracted, yielding 263 distinct source-target pairs. Therefore, the social graph includes 32 vertices and 263 links. In order to capture the quantitative aspects of the collaboration process, we added a weight attribute to each link; this is equal to the number of collaborations between the source student and target student. The range of values taken by the weight was found to be between 1 and 100. A visual representation of the obtained

social graph (Base graph) is included in Figure 3.4.1.

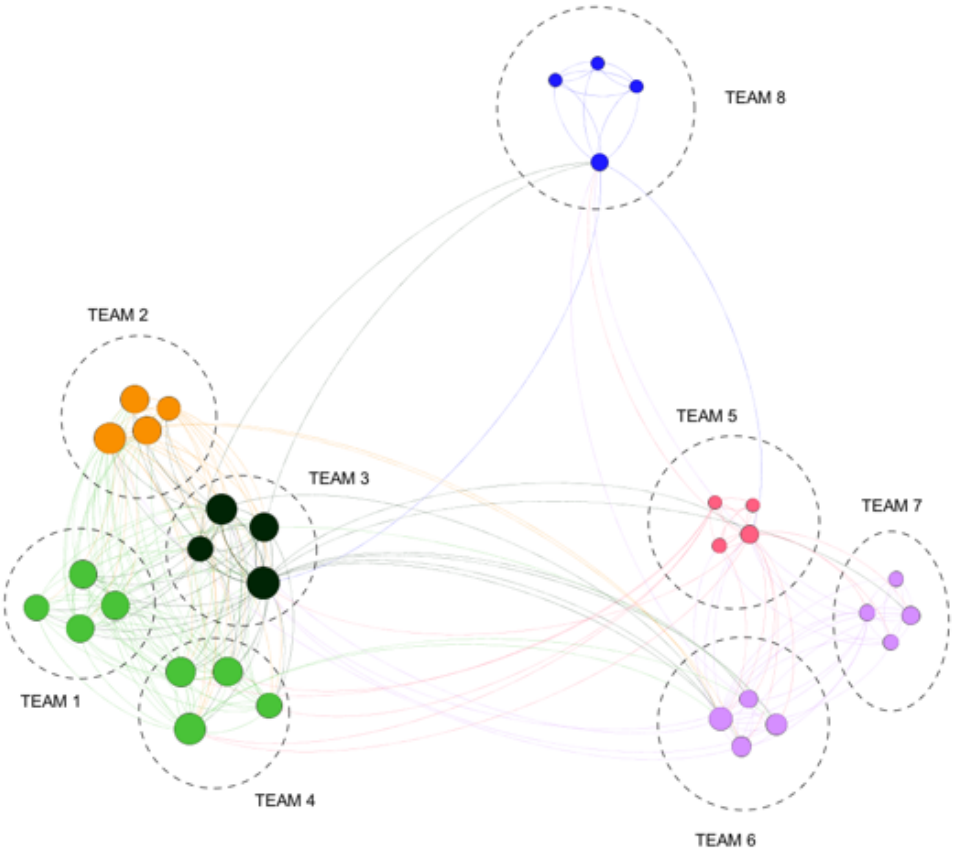


Figure 3.4.1: Graph depicting all social media interactions among students (Base graph). Vertices represent students and links represent collaborations among them. The size of each vertex is proportional to its respective eigenvector centrality value. The colour of the vertices depicts students' clustering as determined by the modularity algorithm. Links are coloured according to their respective source vertex. Graph plotting was done with *Gephi*'s Force Atlas 2 algorithm.

By applying filtering on the *Base graph*, we were able to obtain two subgraphs corresponding to the blog collaborations (as depicted in Figure 3.4.2) and Twitter collaborations (as depicted in Figure 3.4.3). Please note that the union of these two graphs is the *Base graph*. Subsequently, each of these graphs was analysed through various SNA metrics, some of which are presented in Table 3.4.1; all the computations presented in this paper were done with *Gephi* network analysis tool.

In what follows, we discuss the collaboration patterns for each social media tool, as reflected in the metrics and visual representations.

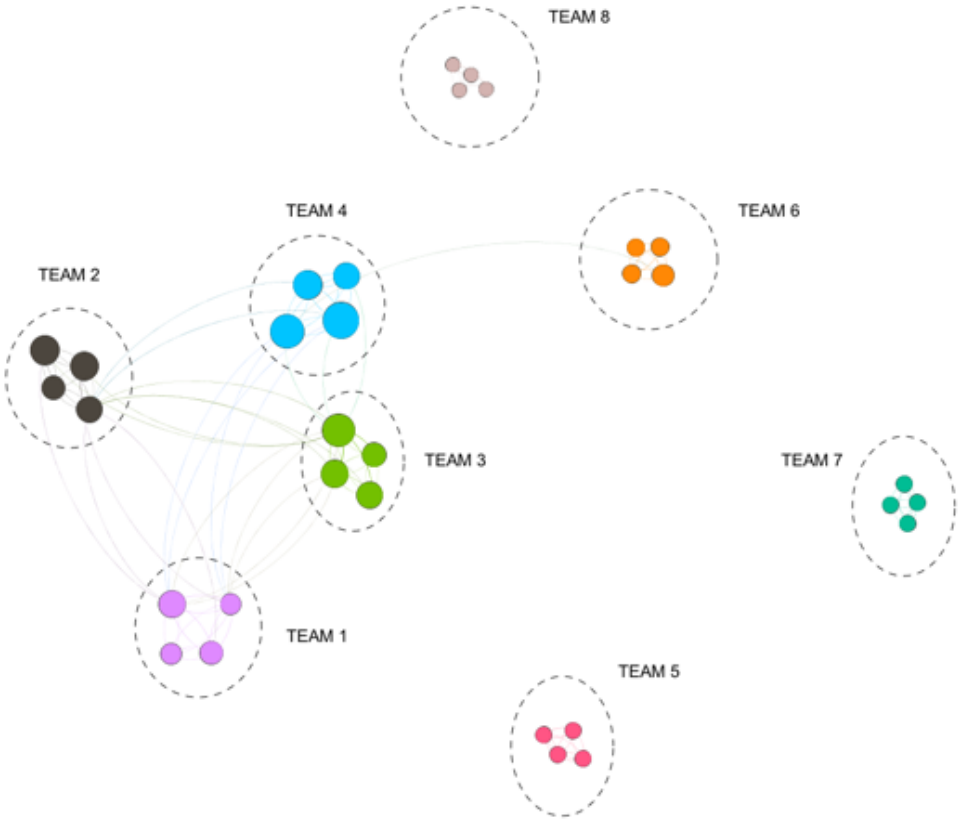


Figure 3.4.2: Graph depicting students' collaborations taking place on the blog (Blog graph). Vertices represent students and links represent collaborations among them. The size of each vertex is proportional to its respective eigenvector centrality value. The colour of the vertices depicts students' clustering as determined by the modularity algorithm. Links are coloured according to their respective source vertex. Graph plotting was done with *Gephi*'s Force Atlas 2 algorithm.

3.4.2 Results & Discussions

As defined in our conceptual framework the social network analysis, see section 3.3.2, is divided in 4 perspectives: whole graph perspective, communities perspective, individual perspective and time-series perspectives. The time series perspec-

Figure 3.4.3: Graph depicting students' collaborations taking place on Twitter (Twitter graph). The size of each vertex is proportional to its respective eigenvector centrality value. The colour of the vertices depicts students' clustering as determined by the modularity algorithm. Links are coloured according to their respective source vertex. Graph plotting was done with *Gephi*'s Force Atlas 2 algorithm.

tive is not present in the current section, as it will be introduced in section 3.5.5. Also, you will find that large majority of SNA methods/techniques identified in the conceptual framework have been used for this analysis. Those few methods/techniques that do appear here have been deamed unnecessary or not applicable in our analysis.

Whole Graph Perspective

As seen in Table 3.4.1, the *Twitter* graph and *Base graph* have very similar results. This is especially due to the fact that 96% of the collaboration links present in the

Table 3.4.1: SNA directed whole-graph metrics. No. communities is determined through the modularity algorithm.

Metric	Base graph	Twitter graph	Blog graph
No. links	263	253	117
Avg. degree	16.43	15.81	7.31
Avg. weighted degree	69.31	51.09	18.21
Diameter	5	5	5
Avg. path length	2.13	2.13	2.21
Density	0.26	0.25	0.11
Modularity	0.49	0.43	0.74
Avg. clustering coefficient	0.65	0.63	0.73
No. WCC (weakly connected components)	1	1	4
No. SCC (strongly connected components)	1	2	6
No. communities	6	6	8

Base graph are also present in the Twitter graph (see No. links metric). By comparison, the *Blog graph* contains only 44% of the total collaboration links, positioning Twitter as a more comprehensive tool for collaboration. The *Avg. degree* metric stands for the number of peers a student has collaborated with (on average). The results depict that students engage in collaborations with other team members also, since a value of 3 would have been an indication of intra-team collaboration only. As shown by the *Avg. Weighted Degree* metric, collaborating students exchange multiple messages, with an average of more than 4 with each peer. Moreover, students engage in collaborations with many more peers on Twitter than on Blog. The *Diameter* and *Avg. path length* metrics represent markers of information diffusion in the social graph. For subsequent information on the choosing of these metrics refer to section 3.3.2 of the conceptual framework.

If we consider the complex networks' classification method introduced in paper [22], together with the current attained metrics' values, we can state that the Base and Twitter graphs are of type *core-periphery*. In our scenario, this categorisation represents a clear evidence of a learning environment that nurtures information exchange.

Density indicates the tendency of students to collaborate with peers from other teams. The results show that Twitter better supports such collaboration patterns, as compared to the blog. *Modularity* and *Avg. clustering coefficient* address the division of communities that arise in social graphs. A higher modularity (range $[-1/2, 1]$) depicts dense collaborations among community members and sparse

collaborations between members of different communities. Graphs with high *Avg. clustering coefficient* (range [0, 1]) characterize inner communities that are close to forming complete graphs/cliques. As it can be seen in Table 3.4.1, the Blog provides better support for this type of collaborations. Furthermore, *No. WCC* and *No. SCC* also depict blog collaborations as intra-team interactions, i.e., highly coupled communities of collaboration arise on the blog. By comparing Figures 3.4.1, 3.4.2 and 3.4.3 we can easily notice that blog collaborations are mainly established among members of the same team, as intended through the instructional scenario. Conversely, *No. communities* metric shows that collaboration among teams is better supported by Twitter, i.e., some teams reach a high level of collaboration and act as one larger community, as seen in Figure 3.4.3.

Based on the above discussion, we argue that both social media tools provide adequate support for collaboration; Twitter supports both inter-team and intra-team interactions, while Blogger mainly supports intra-team collaboration. Moreover, both tools seem to meet the educational purposes intended in the instructional scenario, as mentioned in the second section.

Communities Perspective

In what follows, we discuss in more detail the intra-team and inter-team collaboration patterns, while highlighting salient students and teams. For more details on the methods used see section 3.3.2 of the conceptual framework. By using a *reduction transformation* of the *Base graph* (such that vertices depicting students are aggregated into their respective team vertex, while keeping inter-teams links), we were able to construct the *Teams graph*, as shown in Figure 3.4.4. Hence, we observe two large communities of collaboration: the first is formed by teams 1 to 4 (red community), while the second is formed by teams 5 to 8 (green community). This can be explained by the fact that teams in the red community were scheduled in a different face-to-face session with the instructor than teams in the green community. This means that teams $1-4$ attended each others intermediate presentations, but not the presentations of teams $5-8$ (and the other way around); hence, closer ties were formed inside each community. No clear explanation arose regarding the lower level of inter-team collaboration inside the green community,

compared to the red community; it should be mentioned however that students in the green community had lower average marks than students in the red community. Thus, we argue that various external factors can influence the collaboration patterns.

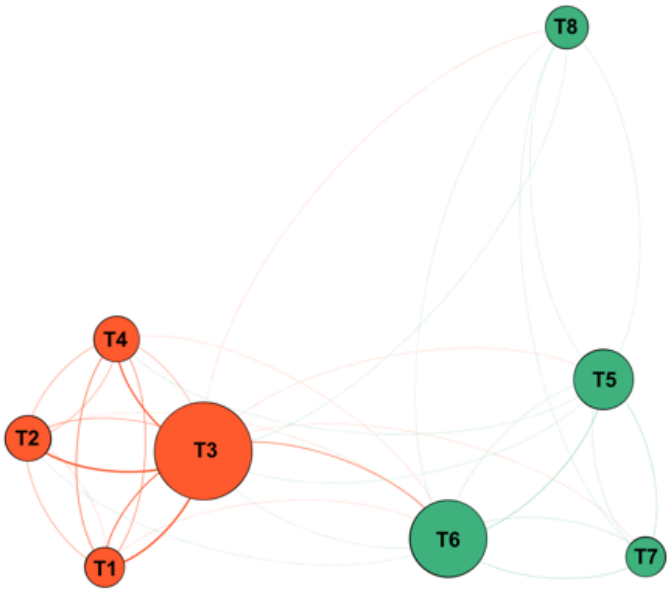


Figure 3.4.4: Graph depicting collaborations established between the teams, both on blog and Twitter (*Teams graph*). Vertices represent teams and links represent inter-team collaborations. The size of each vertex is proportional to its respective betweenness centrality value. The colour of the vertices depicts teams' clustering as determined by the modularity algorithm. Links are coloured according to their respective source vertex. Links' thickness is proportional to their respective weight.

With the exception of Team 8 members, all students engage in inter-team collaboration, each team having two highly active (dominant) students. Sample graphical representations for Team 3 and Team 8 can be seen in Figure 3.4.5 and Figure 3.4.6 respectively.

Nevertheless, the majority of collaborations (1598 out of 2224 or 72%) take place among members of the same team. These results were expected, as intra-team collaboration was required by the instructors, while inter-team collaboration

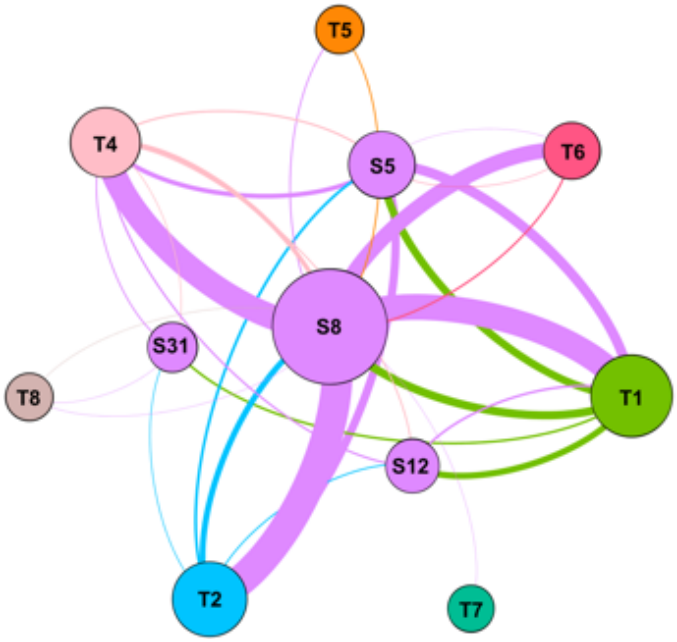


Figure 3.4.5: Inter-team collaborations of students from Team 3. Vertices S5, S8, S12, S31 represent Team 3 students, while the other vertices represent teams (T1, T2, T4-T8). Links depict collaborations among vertices and are coloured according to their source vertex. The size of each vertex is proportional to its respective weighted degree centrality. Links' thickness is proportional to their weight. Graph plotting was done with *Gephi*'s double circular layout.

was only recommended. Hence, the difference can be explained by the pedagogical scenario and the corresponding instructors' actions. Furthermore, we found that all teams display consistent collaboration among their members, especially for the *red community*. Also, every team has two dominant members for this type of collaboration, the same students that are dominant for the inter-team collaborations (except Team 8). Graphical representations of the intra-team collaborations for Teams 3 and 8 can be seen in Figure 3.4.7 and Figure 3.4.8 respectively.

By studying the intra-team and inter-team collaboration patterns, we were able to discover salient teams (T_3 & T_8) and students (S_8 & S_{15}). These teams and students were chosen as they stand on opposite sides of the collaboration spectrum and play key roles in the overall collaboration environment, which we will further

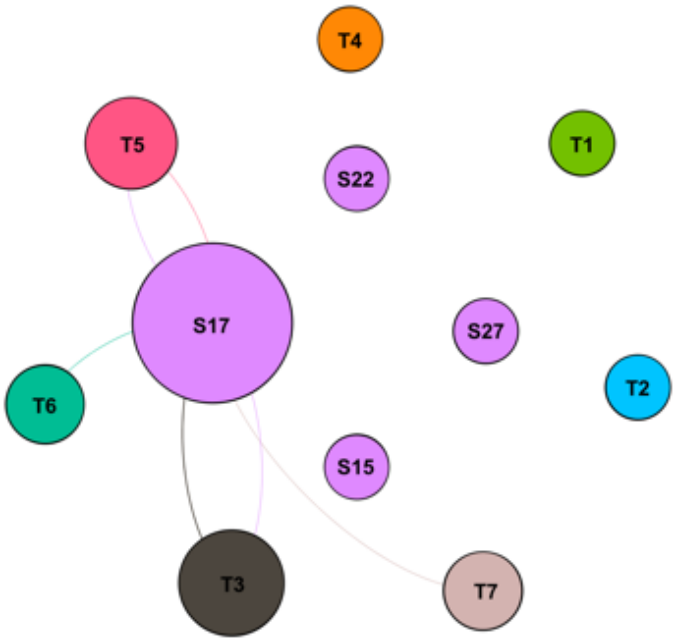


Figure 3.4.6: Inter-team collaborations of students from Team 8. Vertices S15, S17, S22, S27 represent Team 8 students, while the other vertices represent teams (T1-T7). Links depict collaborations among vertices and are coloured according to their source vertex. The size of each vertex is proportional to its respective weighted degree centrality. Links' thickness is proportional to their weight. Graph plotting was done with *Gephi*'s double circular layout.

Figure 3.4.7: Intra-team collaborations for Team 3. Vertices represent students and links collaborations among them. The size of each vertex and is proportional to its respective weighted degree centrality. Links are coloured according to their source vertex and links' thickness is proportional to their weight. Graph plotting was done with *Gephi*'s circular layout.

Figure 3.4.8: Intra-team collaborations for Team 8. Vertices represent students and links collaborations among them. The size of each vertex and is proportional to its respective weighted degree centrality. Links are coloured according to their source vertex and links' thickness is proportional to their weight. Graph plotting was done with *Gephi*'s circular layout.

discuss.

3.4.3 Individual Perspective

In order to discover salient students and teams we employed the graph metrics listed in Table 3.4.2 and Table 3.4.3 respectively. We chose to present here the results for Teams in Table 3.4.3, which as per the conceptual framework should be under the *Communities Perspective*, as these are highly coupled with the results of individual students. This specific metrics were chosen as they are representative for both the topological aspects of the social graph and for the quantitative aspects of the collaborations, see section 3.3.2 of the proposed conceptual framework for more details. *Hubs* and *Authorities* refer in our case to students that initiate many collaborations (source students), respectively to students that are involved in many collaborations initiated by others (target students); i.e., students that share knowledge versus those that receive knowledge. *Betweenness* emphasises students that act as bridges of collaboration, i.e., they facilitate collaboration among peers that otherwise are part of different communities. As seen in Figures 3.4.1, 3.4.4 and 3.4.5, members of Team 3 appear to act as a communication bridge between other teams. Furthermore, in our scenario we interpret *Closeness* as a marker of involvement in the overall collaboration of the respective graph. *Eigenvector* and *PageRank* determine the influence of a student's position in the collaboration graph. *Weighted In-degree* and *Weighted Out-degree* count for the number of

Table 3.4.2: SNA directed graph metrics for vertices, SX stands for Student X. Closeness on the Blog graph was considered only for the giant component. All computations were done with *Gephi* network analysis tool.

	Base graph				*Twitter graph*				*Blog graph*			
Metric	#1	#2	#3	#4	#1	#2	#3	#4	#1	#2	#3	#4
Authority	S9	S8	S10	S11	S8	S10	S9	S12	S10	S8	S9	S5
Hub	S8	S11	S7	S5	S8	S11	S7	S5	S11	S3	S8	S1
Betweenness	S8	S17	S32	S16	S8	S17	S32	S16	S8	S9	S11	S1
Closeness	S8	S32	S11	S5	S8	S32	S11	S5	S8	S1	S3	S11
Eigenvector	S8	S9	S11	S5	S8	S5	S9	S11	S10	S9	S8	S4
PageRank	S8	S15	S5	S16	S8	S9	S1	S5	S10	S16	S5	S7
Weighted In-Degree	S5	S8	S12	S31	S8	S5	S12	S31	S31	S5	S12	S6
Weighted Out-Degree	S8	S5	S1	S11	S8	S5	S1	S32	S8	S5	S11	S15

Table 3.4.3: SNA directed graph metrics for vertices, TX stands for Team X. Closeness on the Blog graph was considered only for the giant component. All computations were done with *Gephi* network analysis tool.

	Teams graph			
Metric	#1	#2	#3	#4
Authority	T6	T3	T4	T1
Hub	T3	T5	T6	T1
Betweenness	T3	T6	T5	T2
Closeness	T3	T6	T5	T2
Eigenvector	T3	T6	T8	T5
PageRank	T3	T4	T1	T2
Weighted In-Degree	T4	T2	T1	T3
Weighted Out-Degree	T3	T1	T4	T2

collaborations received, respectively for the number of collaborations initiated by a student/team. Highly ranked students and teams are included in Table 3.4.2 and Table 3.4.3.

As mentioned in section 3.4.2 and also shown in Table 3.4.2, student *S8* can be considered as the top ranked student (collaboration-wise). In order to explain her almost unchallenged top ranking we conducted further investigations. University records show she was involved in two long-duration foreign exchange projects in different locations, which could suggest a highly social person. Her peers confirmed that *S8* is indeed a very sociable and talkative learner. By conducting an interview with the student, we determined that she was actively engaged in social media, managing a globetrotter blog. Furthermore, she has participated in formal

and informal training on social media techniques. Hence, we reached a potential explanation of the high collaboration rankings achieved.

Moreover, student *S8* is part of team $T3$, which can also be considered as the leading team from a collaboration point of view. The importance of this team can be explained by two factors. First, by looking at *Weighted Out-Degree* metric in Table 3.4.3, we can see that students in $T3$ initiated the highest number of collaborations; however, the team is not placed in the top 3 according to *Weighted In-Degree* metric. Hence, we argue that $T3$ is actively sharing more knowledge than it receives, supposition confirmed to some extent by Figure 3.4.5. Second, the other metrics in Table 3.4.3 show that $T3$ is situated at the very centre of the collaboration paths, as can be seen also in Figure 3.4.4. Hence, the quantitative and graph positioning factors make this team the highest ranked collaboration-wise.

3.5 StudentViz: A Tool for Visualizing Students' Collaborations

Thus far in the current chapter, we have proposed a conceptual framework for knowledge extraction and visualisation based on SNA in section 3.3 and paper [24], which we have validated in section 3.4 and paper [25]. *Gephi* network analysis tool was used for all computations and graph visualisations, which adequately fitted researcher's needs; however, the tool was deemed too complex for instructors, who are not specialists in SNA or visualisation.

Therefore, we decided to build a simple network visualisation tool, easy to use by the teachers and specifically designed to work in conjunction with *eMUSE* social learning environment. The tool should provide useful and relevant visualisations from the instructor's point of view, therefore we first identified a list of visualisation needs (VN) outlined by the teachers working with *eMUSE*:

VN1 Visualise the general status of collaboration.

VN2 Visualize the status of collaboration for each community.

VN3 Visualize the status of collaboration for each learner.

Furthermore, the tool should support the processes of gaining insight through information visualisations, as identified in [233]:

P1 Provide Overview - grasp the big picture of a dataset.

P2 Adjust - explore a dataset by changing the abstraction level or selection range (e.g., by filtering, grouping, aggregating).

P3 Detect Pattern - find relationships, trends, or anomalies in the dataset.

P4 Match Mental Model - correlate the data with the user's mental model of it, in order to facilitate understanding.

Starting from these requirements, we designed and implemented our *StudentViz* tool, as described further.

3.5.1 Related Visualisation Platforms

Many network visualisation platforms (NVP) are available [46]; however, they are not specifically built for educational settings, so we wanted to investigate whether they can be used for our particular learning scenario, in conjunction with *eMUSE* platform. As our target users are instructors with limited technical expertise, we imposed some initial restrictions. The considered tools should be free, easy to install and use, and operating system independent. Moreover, they should provide high flexibility, so that instructors could adapt the visualisation methods to their needs. Hence, our evaluation included the following platforms:

- Cuttlefish[10] focuses on ease of use, but with limited capabilities and no flexibility; we also experienced some visualisation glitches upon using the zooming functionality.
- Cytoscape is an open source platform developed for molecular networks visualisation, that has expanded its use across various network related research fields [200]. Its standard features are relatively easy to use. However, the platform lacks flexibility of the visualisation methods.
- Visione[11] has similar capabilities with Cytoscape, but it provides even less

[10] http://cuttlefish.sourceforge.net/
[11] https://visone.info/

flexibility and the user interface is cluttered and non-intuitive.

- Tulip is an NVP that focuses on relational data [9]. It provides highly flexible visualisation and a wide range of analysis capabilities for various research fields.

- Gephi aims to be a general platform of analysis and visualisation for all kinds of networks [19]. Its clear design and resemblance with Photoshop make it very easy to use. Furthermore, its visualisation capabilities are flexible and extensible through plugins.

Overall, we found *Gephi* and *Tulip* to be equally capable in terms of visualisation functionalities; however, *Gephi* provided a better user experience. Nevertheless, all platforms were considered too complex by the instructors, including many irrelevant functionalities for their purpose and requiring SNA expertise. Also, some of the desired visualisations (e.g., team and community perspectives) required significant effort in order to be generated with the existing NVP, starting from our available *eMUSE* dataset. Therefore, we decided to build a network visualisation tool dedicated for the teachers, with a simple and intuitive interface, as described next.

3.5.2 Data Acquisition & Graph Building Module

StudentViz is a data visualisation tool purposely built to work in conjunction with *eMUSE* social learning environment. Its aim is to provide suggestive visualisations of students' collaboration patterns, as they are recorded by the platform. All students' social media traces are monitored and stored by *eMUSE*, and *StudentViz* uses these data to draw the graphs depicting social media interactions between the students. A schematic representation of the data flow is shown in Figure 3.5.1.

Based on the conceptual frameworks principles of constructing the social graph, see section 3.3.1, and previous experience with building social graphs from *eMUSE* data, see section 3.4.1, a data acquisition & graph building module (denoted DtoG) was developed to bridge the gap between the data source (*eMUSE*) and the visualisation tool (*StudentViz*). DtoG processes the raw data collected by *eMUSE*, filter-

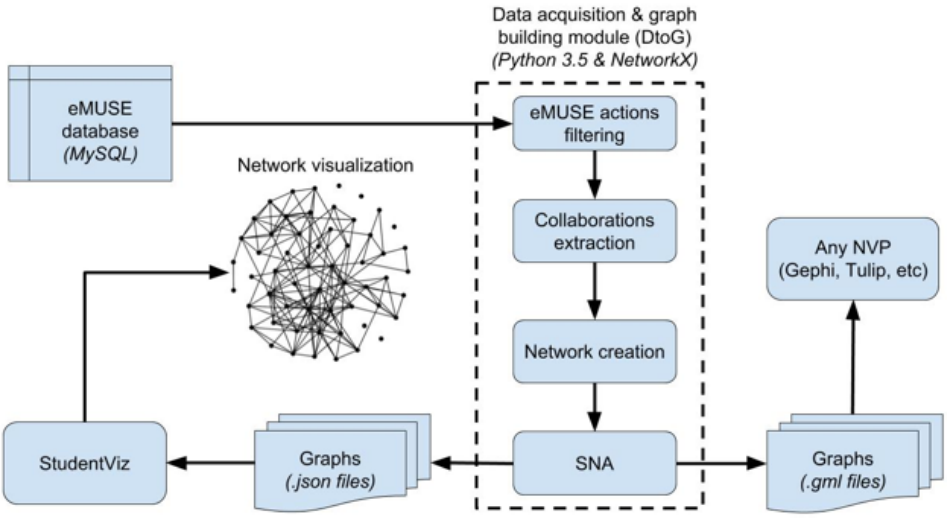


Figure 3.5.1: StudentViz - network visualisation data flow.

ing the collaboration actions, and then creates various social graphs on which several SNA methods are applied. More specifically, directed graphs are built starting from students' interactions on the Blog and micro-blogging tool (Twitter); nodes represent learners and links represent messages exchanged through the social media tools integrated in *eMUSE*. These graphs can be exported in various formats (e.g., .gml or .json files), which can be subsequently input into any NVP, including *StudentViz*. As far as implementation is concerned, DtoG was built using Python 3.5 programming language and *NetworkX* graph analysis library.

As part of the the DtoG module several edge/link weighting methods were developed in order to capture collaboration from various facets. The first method *wNC*, see Equation 3.1, computes the weight of an edge as the number of collaborations between two students. Although, this is the most usual method used in the academia it fails to capture the qualitative aspects of collaboration, which we will try to capture through the subsequent equations. From our perspective, the re-tweeting mechanism (RT) is used to emphasise a useful piece of information, therefor collaborations containing such a mechanism should receive a heavier weight, see Equation 3.2. *wRT* method rewards the student that re-tweets the message, he/she is perceived as a person that strengthens collaboration through reviewing a message content and passing it along. Regarding the *wLK* method,

see Equation 3.3, students consider sharing internet links as an efficient manner of collaboration; it requires minimal time effort, from the perspective of the contributor, and high input of information, from the perspective of the recipient. Further more, this sharing method is suitable for Twitter due to the message length restriction. The *wALL* method encapsulates both qualitative and quantitative aspects of the collaborations in case. The values of α and β were experimentally deducted with a higher emphasis on students that review and share information; hence the higher value of α. Note that the wRT method only makes sense when Twitter collaborations are taken in consideration.

$$wNC_{edge(A,B)} = No.collaborations(A, B) \tag{3.1}$$

$$wRT_{edge(A,B)} = \sum_{n=1}^{No.collaborations(A,B)} collaboration_n * \alpha \tag{3.2}$$

$$wLK_{edge(A,B)} = \sum_{n=1}^{No.collaborations(A,B)} collaboration_n * \beta \tag{3.3}$$

$$wALL_{edge(A,B)} = \sum_{n=1}^{No.collaborations(A,B)} collaboration_n * \alpha * \beta \tag{3.4}$$

$$\alpha = \begin{cases} 1 & \text{if } collaboration_n \text{ is not RT} \\ 3 & \text{if } collaboration_n \text{ is RT} \end{cases}$$

$$\beta = \begin{cases} 1 & \text{if } collaboration_n \text{ does not contain an internet link} \\ 2 & \text{if } collaboration_n \text{ contains an internet link} \end{cases}$$

Collaborations present in the *Base graph*, see section 3.4.1, can be filtered by their source, established on Twitter or Blogger. Hence, we can construct a collaboration graph for each social media platform and compare their traits. Also, we

considered applying the two graph transformation methods: *reduction* and *context*; which we have discussed in section 3.3.2.

To be able to conduct temporal analysis, we need to construct graphs of collaborations established during specific time frames. We established the length of a time frame to be 2 weeks, as students met with the instructors every other week. Hence, such a time frame is adequate to see progress between instructor-student encounters. The construction of temporal graph is identical with the construction of the *Base graph*, after restricting the set of collaborations to those established during the specified time frame. As the course spans for 14 weeks there are 7 non-overlapping time frames. However, we need to consider the two weeks of winter vacation, as student may still be active. We will further mention them as TFx, x stands for the starting week. Hence, TF1 is the time frame starting with day one of the first week and ending with day seven of the second week. Similarly, TF7 is a 2-week time frame starting with day one of week 7. In order to address the time frame of the entire semester we will mark it as Tall.

We developed the following graph naming schema: *source_weighting-method _timeframe_[transformation-method]*; as a concise and clear way of describing graphs. The source can be: twitter, blogger or all; standing for the social media source of the collaborations, all stands for both twitter and blogger collaborations. See Equations 3.1 to 3.4 for the possible weighting methods. The transformation method can be: *reduction*, *context[team]* or none. Where the *team* parameter in *context[team]* represents the ID of a *eMUSE* team. Upon the use of *context[team]*, a context transformation of the attained graph with the focus on a specific *eMUSE* team shall be applied, e.g., context2. For example, the graph *twitter_wALL_T5_context3* is a graph forged from messages gathered on Twitter during week 5 and 6. Further more, the edge weighting method *wALL* takes in consideration re-tweets and the presence of internet links while putting an emphasis on team 3. The scope of forging this graph is to allow instructors to analyse the collaboration activity on Twitter for students in team 3 during week 5 and 6, with a special regard for those that review and share information as internet links. It is noteworthy that the DtoG module generates automatically all possible graphs, the instructor needs only to run the module and understand the naming schema.

3.5.3 StudentViz Design

General Design

In an attempt to reduce instructors' effort in using *StudentViz*, we decided to conceive it as a web application, thus eliminating the need of installation, configuration and manual updates. PHP, HTML5, CSS and *Cytoscape*[12] JavaScript library were used for implementing the tool.

We also discussed with the instructors in order to agree on a set of graph plotting conventions that would be most suitable for their needs. The visualisation methods should be easily correlated with humans' mental map (insight gaining process P4 [233]), thus reducing the comprehension effort. Therefore, we used directed graphs, in which nodes represent learners and links represent messages sent between the learners (on Blog or Twitter). In order to expand the dimensionality of the information rendered in the graph, we introduced a colour schema and magnitude schema for each graph element. Nodes shall be coloured according to their affiliation to a certain community, i.e., nodes representing learners of the same team / community shall have the same colour. In addition, links shall be coloured according to their source node, in order to represent link direction. For example, if student A (red-coloured node) sent a message to student B (green-coloured node), then the link between nodes A and B shall be coloured in red. The magnitude of each node (i.e., diameter) shall be directly proportional to a chosen SNA ranking: the larger the node, the higher the ranking. Thus, instructors can easily compare students according to a selected SNA ranking method, e.g., *PageRank* [173]. In order to map the node to a particular learner, nodes shall be labelled with a unique learner ID. Furthermore, the thickness of each link shall be directly proportional to the strength / intensity of the collaboration between the two students; as computed described in Equations 3.1 to 3.4.

We also decided to use force directed methods (FDM) for graph plotting [88], which generally produce aesthetically pleasing results. These methods are based on attractive and repulsive forces inspired from physics. Such forces attract nodes with high connectivity and repulse those with low connectivity, making the observation of communities of collaboration very intuitive. Moreover, the distance

[12] http://js.cytoscape.org/

among nodes is inversely correlated with the strength of their influence on each other. Another advantage of FDM is their adaptability to various network traits, so they can be optimised from case to case.

Visual Design

StudentViz interface was designed in a simple and intuitive way. Two views are available, similar to *Gephi*: the *Main view* and the *Data view*. The *Main view* is further divided into 3 areas: *Options area* (left side), *Plotting canvas area* (centre) and *Additional information area* (right side). Figure 3.5.2 provides a screenshot of *StudentViz Main view*, see Figure 3.5.3. The *Data view* consists of a sortable grid of learners' attributes, including various SNA metrics. Similar views can also be found in *Gephi, Cytoscape* or *Tulip* under various names (e.g., Data laboratory in *Gephi*).

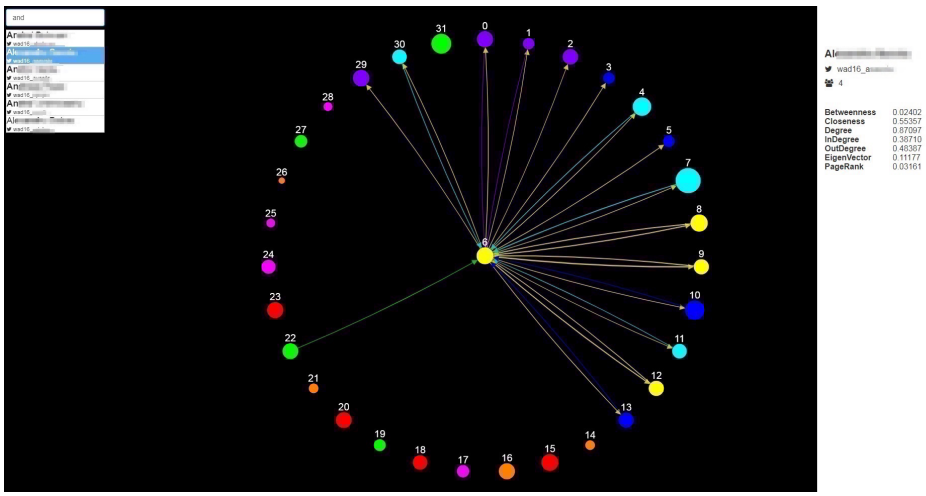
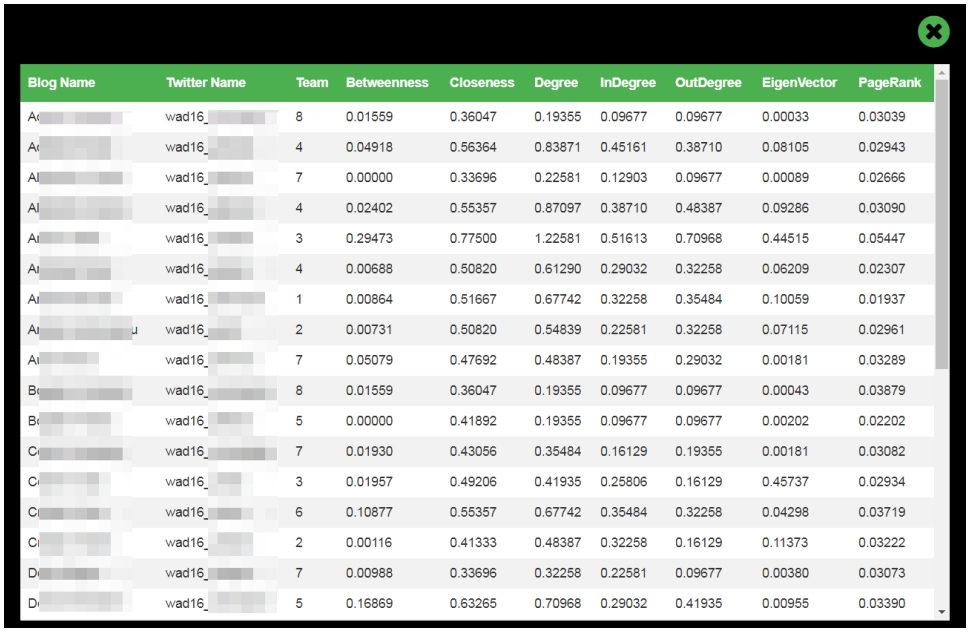


Figure 3.5.2: StudentViz Main view - Focus-circular layout is employed; nodes' diameters are proportional to their PageRank, while their colours depict affiliation to a specific team.

In what follows we present the *Main view* in more detail. The *Options area* allows instructors to interact with the visualisations and adjust them through various settings. Thus, as collaboration cannot be quantified by just one SNA metric, the teacher has the possibility to choose from several metrics: *Betweenness, Close-*



Blog Name	Twitter Name	Team	Betweenness	Closeness	Degree	InDegree	OutDegree	EigenVector	PageRank
A	wad16_	8	0.01559	0.36047	0.19355	0.09677	0.09677	0.00033	0.03039
A	wad16_	4	0.04918	0.56364	0.83871	0.45161	0.38710	0.08105	0.02943
Al	wad16_	7	0.00000	0.33696	0.22581	0.12903	0.09677	0.00089	0.02666
Al	wad16_	4	0.02402	0.55357	0.87097	0.38710	0.48387	0.09286	0.03090
Al	wad16_	3	0.29473	0.77500	1.22581	0.51613	0.70968	0.44515	0.05447
Al	wad16_	4	0.00688	0.50820	0.61290	0.29032	0.32258	0.06209	0.02307
Al	wad16_	1	0.00864	0.51667	0.67742	0.32258	0.35484	0.10059	0.01937
Al	wad16_	2	0.00731	0.50820	0.54839	0.22581	0.32258	0.07115	0.02961
Al	wad16_	7	0.05079	0.47692	0.48387	0.19355	0.29032	0.00181	0.03289
Be	wad16_	8	0.01559	0.36047	0.19355	0.09677	0.09677	0.00043	0.03879
Be	wad16_	5	0.00000	0.41892	0.19355	0.09677	0.09677	0.00202	0.02202
Ch	wad16_	7	0.01930	0.43056	0.35484	0.16129	0.19355	0.00181	0.03082
Ch	wad16_	3	0.01957	0.49206	0.41935	0.25806	0.16129	0.45737	0.02934
Ch	wad16_	6	0.10877	0.55357	0.67742	0.35484	0.32258	0.04298	0.03719
Ch	wad16_	2	0.00116	0.41333	0.48387	0.32258	0.16129	0.11373	0.03222
Da	wad16_	7	0.00988	0.33696	0.32258	0.22581	0.09677	0.00380	0.03073
Da	wad16_	5	0.16869	0.63265	0.70968	0.29032	0.41935	0.00955	0.03390

Figure 3.5.3: StudentViz Data view.

ness, Degree, In-Degree, Out-degree, Eigenvector and *PageRank*. Through *Betweenness* an instructor can determine the students that bridge community silos, those that facilitate the exchange of knowledge between communities. Students with high *Closeness* values are positioned on various communication paths, playing an important role in knowledge diffusion. *Degree, In-Degree* and *Out-Degree* centrality metrics can be used to determine the most / least active learners. Both *Eigenvector* and *PageRank* are measures of nodes importance that take into consideration qualitative and quantitative aspects; an important student is defined as one that has multiple collaborations with other important students. Additional information about these centrality metrics can be found in [34].

Another functionality provided in the *Options area* allows the instructor to select the graph plotting algorithm; available choices are: *WebCola*[13], *Cose-Bilkent* [67], *Circular* and *Focus-circular*, which will be further discussed. Furthermore, the instructor can also choose the focus of the visualisation: individual learners, teams or communities. This functionality is achieved by applying a *reduction transformation* on graphs that include all students; learners of the same team are repre-

[13] http://marvl.infotech.monash.edu/webcola

sented as one node, while filtering out intra-team collaborations. Furthermore, the nodes' colour can depict team or community affiliation; teams are predetermined from the beginning of the semester, while communities are non-formal and self-regulated. Community detection is computed using a Laplacian method [131].

An additional option available to the instructor is to load various graphs created by the DtoG module (e.g., graph containing all social media interactions among students, graph containing only collaborations on the Blog / Twitter). Finally, for easy identification of each student / team, an auto-complete search box is provided, in addition to the full list of students.

The central area of the *Main view* consists of a black canvas on which the graph is plotted. The canvas colour was chosen in order to provide a high contrast for the graph nodes and edges. The instructor can re-position nodes through drag-and-drop functionality; he can also select one node for detailed inspection, which sets the graph plotting algorithm to *Focus-circular* and opens the *Additional information area*.

Finally, the right side area of the *Main view* provides information regarding the specific node selected: student name, team, SNA metrics values. This area is only displayed upon selection of a node, otherwise it is hidden, to allocate a larger space for the plotting canvas

3.5.4 Illustrating Visualisation Functionalities in StudentViz

In what follows we show how *StudentViz* answers instructors' visualisation needs, as they were specified in the Introduction (VN1 - VN3). It also provides support for the general processes through which people gain insight when using an information visualisation system (P1 - P4) [233].

The context of use is a course on Web Applications Design, taught to 4th year undergraduate students from the University of Craiova, Romania, during 2016-2017 winter semester. 32 students used *eMUSE* platform (and the associated social media tools) for communication and collaboration support. Based on the social media traces collected by *eMUSE*, a total of 2224 collaboration links were extracted (263 having distinct source-target pairs). Therefore, a social graph with 32 vertices and 263 links was built.

Thus, Figures 3.5.4 and 3.5.5 provide a birds-eye view on learners' collaboration, by using *Cose-Bilkent* and *WebCola* FDM respectively. These algorithms have low computational workload in case of small graphs and are suited for interactive applications, as they avoid the overlapping of nodes. As seen in Fig. 3 and 4, the general pattern of collaboration can be easily spotted, thus supporting VN1 and P1. High and low density areas of collaboration can be easily identified in both figures. Learners in the high density area (with nodes tightly plotted together) are significantly involved in collaboration with members of diverse teams; students in low density areas are those that teachers should focus on in order to enhance collaboration.

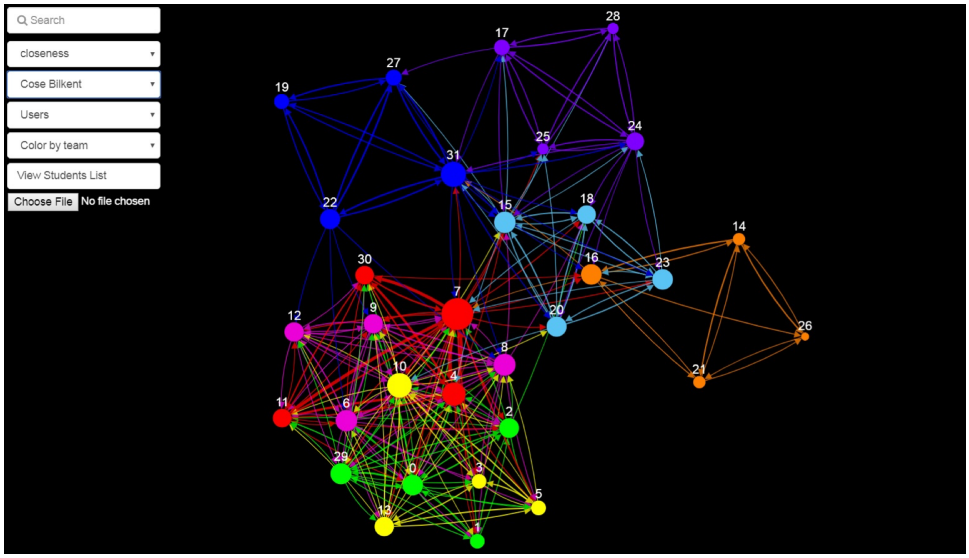


Figure 3.5.4: Visualisation provided by *StudentViz* using *Cose-Bilkent* plotting method. Nodes' diameters are proportional to *PageRank* metric, while their colours depict affiliation to a specific team.

Although both algorithms produce similar visualisations, there are some variations that justify their complementary use. *WebCola* favors the identification of large communities of collaboration, as nodes are plotted in close proximity. However, this creates clutter, making smaller communities (teams) hard to spot. In turn, *Cose-Bilkent* favors the observation of smaller communities over large ones. Hence, *WebCola* and *Cose-Bilkent* also support VN2. In addition, these methods

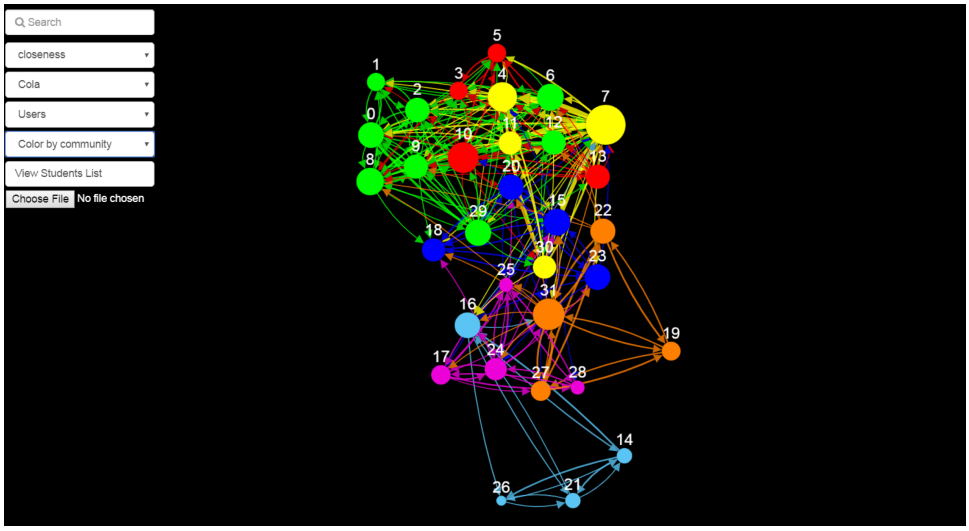


Figure 3.5.5: Visualisation provided by *StudentViz* using *WebCola* layout. Nodes' diameters are proportional to *PageRank* metric, while their colors depict affiliation to a specific community.

allow the discovery of the general structure and trends of collaboration, thus addressing P3.

Furthermore, instructors can better assess the collaboration between the teams by visualising teams as nodes, like in Figure 3.5.6 (which was obtained by applying the graph *reduction transformation*). Both Figure 3.5.2 and Figure 3.5.6 are illustrations of the *Focus-circular* plotting method, devised to clearly observe the status of a particular node (learner or team, respectively). The node of interest is positioned in the center of a circle on which the other nodes are plotted. Moreover, only the collaborations that involve the node of interest are rendered, to reduce unnecessary clutter and allow the instructor to focus on the particular node. Hence, these focus-circular visualisations address both VN2 (when nodes represent teams) and VN3 (when nodes represent students). Furthermore, P2 is also supported, as instructors can change the perspective on the data-set by selecting the node of interest.

Finally, the *Circular* plotting method, inspired by *Gephi*'s circular layout, was devised to better observe the status of each team, as illustrated in Figure 3.5.7. First, an average value of the selected SNA metric is computed for each team. The team

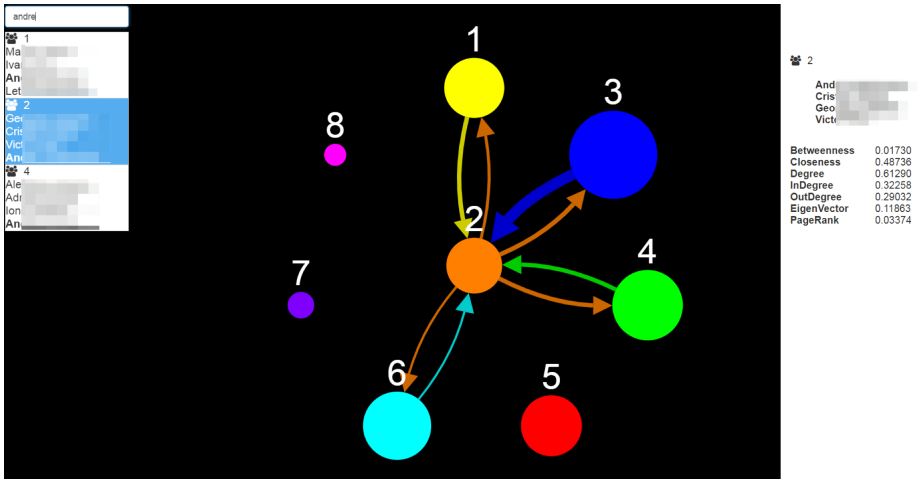


Figure 3.5.6: Visualisation provided by *StudentViz* using *Focus-circular* layout. Nodes represent teams and their diameters are proportional to *PageRank* metric.

with the highest ranking is plotted first, followed by the other teams according to their average metric rank, in a clockwise descending order. Individual nodes are also rendered in a clockwise descending order in the designated plot area for their team, according to their SNA metric value. Hence, the first plotted node depicts the learner with the highest selected metric value from the highest ranking team. This visualisation method allows the teacher to observe the status of each team in comparison with other teams, but also the status of each learner in comparison with his fellow team members. Thus, both VN2 and VN3 are addressed by this visualisation method; moreover, P3 is supported here, as instructors can discover collaboration patterns among teams and students.

3.5.5 Insights obtained through StudentViz

Through the use of *StudentViz* we were able to gather valuable insights, out of which we will discuss some. By presenting these insights and their manner of discovery we hope to provide better evidence on the utility of our proposed tool. These insights refer to the data in question, however we argue that similar or even additional insights could be obtained no matter the data/context.

As mentioned before, *StudentViz* also has a *Data view*, which a tabular rep-

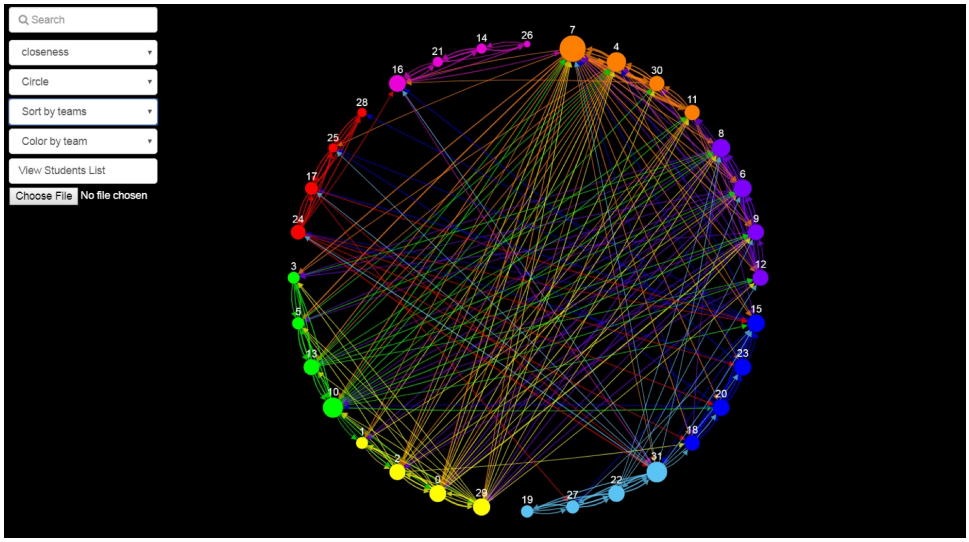


Figure 3.5.7: Visualisation provided by *StudentViz* using *Circular* layout. Nodes' diameters are proportional to *PageRank* metric, while their colours depict affiliation to a specific team.

resentation of data on each student/team. The data represent numerical values computed for various SNA metrics. Hence, for each student/team the *Data view* presents a comprehensive numerical status. However, the real utility of this view is the ability of compare and rank students/teams according to the SNA metrics considered. Hence, an instructor can observe the student's performance in comparison with others and perceive the magnitude of the differences through the depicted values. We chose to present the *Data view* here as it is a good starting point for each discovering insights, as we shall exemplify in the following lines. In our experience of working with *StudentViz* and other similar platforms, we were able to discover insights by observing some particularities in *Data view* and then confirming them in *Main view* or vice versa.

One of the insights revealed as we were analysing students as part of their respective team. To do so, we ranked the students in *Data view* by Team. Hence, we were able to observe the performance of each team member, compared to each other, and the general distribution of their social metrics' values. An instructor with this insight on teams could devise targeted methods for the under-performing teams. Such teams can also be identified by observing the *Main View*. For example, in Figure 3.5.6. you can observe a graph that depicts communication between

teams, aggregated communication among student of different teams. As the the diameter of the nodes is tied with the *PageRank* metric, the larger the node the more social is the respective team. Hence, another way or a complementary way to observe teams' status.

Through the help of the DtoG module, *StudentViz* is able to provide the instructors with visualisations and data not only for the entire period of time, but for specific time periods also, see Figures 3.5.8 and 3.5.9 for exemplification. Such capability can provide valuable insights on the development of the entire learning/collaboration network. By looking at the inter-teams collaboration patterns that form higher order communities, we observed that teams forming stable communities (over time) have more intense social interactions. Hence, collaboration among teams is good when it is relatively stable/lasting, e.g. teams 1 & 3. In order to sustain our observation, we computed a tabular manner of representing community formation in our scenario, see Table 3.5.1. The table was computed by loading the following graphs: all_t5, all_t7, all_t9 & all_t11; and colouring the nodes based on their Partition. Hence, we could visualise the communities of students that were present in the weeks 5&6, 7&8, 9&10 and 11&12. Than, we included the initial team allocation which could be found in the *Data view* as Team id. In the above mentioned Table we can see that teams 1 & 3 form a community (backbone) from start to finish, hence they collaborate in a stable manner. The community formed by the above mentioned teams is joined in TF3 by team 4, which we'll be sustained also in TF4 with the exception of one student. Team 2 is generally forming it's own community with the exception of TF2, when it was part of the backbone community. Team 4 started on its own and ended as part of the backbone community, with the exception of one student, a case for future work. Hence, we can argue that collaboration among teams 1 – 4 is relatively stable, at least compared with teams 5 – 8. Also note that teams 1 – 4 have the highest *PageRank* values and that visualisations of Figures 3.5.4, 3.5.5 & 3.5.7 clearly show that they collaborate substantially. For Team 5 and Team 8 the inter-team collaboration patterns have similarities, both teams have tried to collaborate with other teams but no collaboration was sustained for more than one time-frame. However there are some substantial differences, team 5 is depicted in the majority of plots as sustaining a closer collaboration with Teams 1-4 than teams 6 – 8, hence

we consider this team to be the most collaborative among teams 5-8. Team 8 is at the opposite spectrum of collaboration, very few collaborations are established outside the team. Team 5 & 6 and try various collaborations with other teams, however no stable communities are formed (lasting at least two consecutive time frames). Based on the above observations we could identify 3 general collaboration patterns for our scenario. Teams that collaborate in a stable manner (teams 1,3,4), teams that collaborate but not in a stable manner (teams 2,5,6,7) and teams that barely collaborate (team 8).

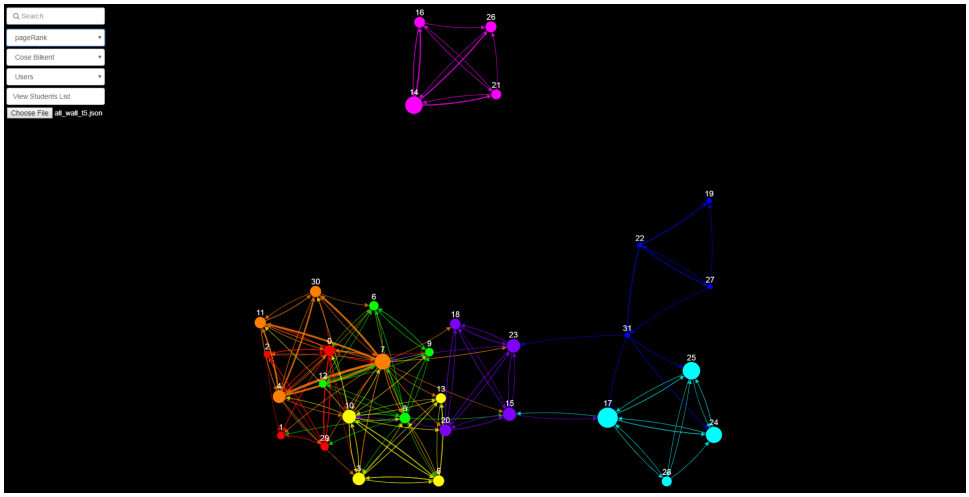


Figure 3.5.8: Visualisation provided by *StudentViz* using *Cose-Bilkent* layout. Nodes represent students and their diameters are proportional to *PageRank* metric, the graphs was constructed only based on the communication established in the 5th and 6th week of the course.

Since the students in our class are able to communicate by using Twitter and Blogger we provide the instructors the possibility to visualise/analyse the collaborations established on each social media tool and aggregated collaborations. T his functionality is provided by the DtoG module and not by *StudentViz* per se. Thus, instructors can evaluate students not only on the entire collaboration spectrum but for specific collaborations also. For our scenario, having insights on how students/teams rank by social media tool has sense, as the tools were introduced to support different types of communication with each having specific purposes.

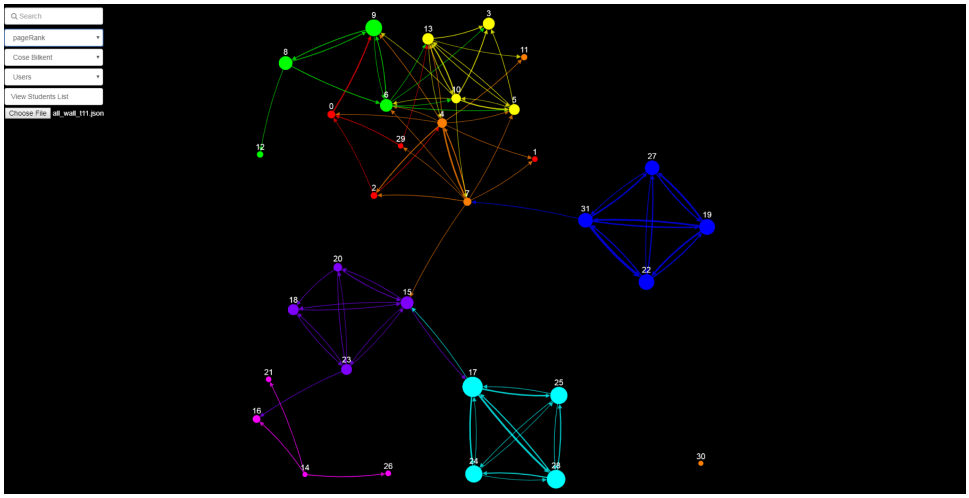


Figure 3.5.9: Visualization provided by *StudentViz* using *Cose-Bilkent layout*. Nodes represent students and their diameters are proportional to *PageRank* metric, the graphs was constructde only based on the communication established in the 11th and 12th weeks of the course

Twitter has the role of support non-formal, fast exchange of communication which mainly consists of opinions and internet links, this was also the preferred communication method of the students. On the other part, Blogger collaborations are more formal since the authors need to write longer and scientifically wise texts. Also, blogger posts were written when key milestones were reached during the projects' development phases. Hence, we consider that a good collaboration between the students should also be reflected both on Twitter and Blogger. As you can see in Figures 10 and 11 the above conclusion is true for some teams, while for other teams only Twitter plays the major role as collaboration tool.

3.6 Conclusions & Future Work

3.6.1 Conclusions on the Analysis of Textual Complexity Indices

The first experiment investigated how students' writing style on social media tools, corroborated with the time evolution of their posts, can be used to predict their academic performance. Textual complexity and longitudinal analyses were performed on the blog and microblog posts of 148 (out of the total 343) students

Table 3.5.1: Table depicting the manner communities arise from collaborations among teams. On the Y axis we represent the teams, while on the X axis we represent 4 consecutive time frames. The *PageRank* column depicts the value obtained by each specific team in the teams graph. For readability reasons the value of the *PageRank* has been multiplied by 100 and rounded. Cell colour depicts the pertinence of at least 3 members in a team to specific community, i.e. the same colour the same community. The letter/letters in the in each cell also depict pertinence to a community, but with additional information. In the case when a cell has one letter, then all the members of that respective team are part of the same community. Otherwise, a number in front of each letter would represent the number of team members that are part of the community specified by the letter, e.g. 3a,1b depicts that 3 members of the respective team are part of community a and one member of community b. Hence, the column positioned on row 3 column TF9 should be read as, all team members from team 3 are part of community a which is also depicted by the colour red.

Team	PageRank	TF5	TF7	TF9	TF11
Team 1	15	a	a	a	a
Team 2	14	b	a	b	b
Team 3	21	a	a	a	a
Team 4	16	c	c	a	3a, 1b
Team 5	8	d	d	d	d
Team 6	13	e	d	g	e
Team 7	10	d	f	g	h
Team 8	3	h	h	h	h

engaged in a project-based learning activity during 6 consecutive instalments of the Web Applications Design course.

The analyses indicated that students who received higher grades in the course had greater word entropy, used more verbs, prepositions, adverbs, and pronouns, produced more unique words, and more named entities. Additionally, students who received higher grades had lower inner cohesion per contribution, indicating more elaborated paragraphs that represented a mixture of different ideas in the context of each contribution. The time series variables denote a more uniform distribution, with weekly fluctuations in terms of participation, which is normal for students that were more actively involved in using the social media tools. Three of these variables (word entropy, time series entropy and average verbs in sentence) were predictive of performance in both a regression analysis and a DFA. The results are promising as several significant correlations and statistical models were

identified in order to predict academic performance (i.e., course grades) based on textual complexity and longitudinal analysis indices. Additional experiments that will consider the learning style of each student, as well as an equivalent textual complexity model for Romanian language, are underway in order to augment the depth of our analyses. This will enable the consideration of a higher sample of students from the total of 343 course participants and will increase the power of the applied mechanisms.

The second experiment investigated how students' writing style on social media tools can be used to predict their academic performance and learning style. Textual complexity analyses were applied on the blog and microblog posts of 27 students engaged in a project-based learning activity. The results are encouraging: several significant correlations were found between textual complexity indices and project / exam grades and one index (average named entities per contribution) was determined to be a significant predictor. Similarly, several indexes were predictive for three of the four FSLSM dimensions: two predictors for the active/reflective dimension, three predictors for the visual/verbal dimension, and one predictor for the sequential/global dimension; no variables were predictive of the sensing/intuitive dimension. As expected, since the students' online participation was directly linked to the project grading, it was normal that the effect size for scoring projects is greater in contrast to the evaluation relying on the final exam grades.

It should be noted that the presented experiment was only a pilot study with a small sample size; therefore, its power is low. Nevertheless, the employed mechanisms are extensible and can be easily applied on a larger student population.

3.6.2 Conclusions on the use of SNA methods

As part of this chapter, we have introduced a conceptual knowledge extraction framework to explore the collaborative learning process supported by *eMUSE* environment and the integrated social media tools. A set of knowledge extraction methods based on SNA techniques were proposed, starting from specific educational needs identified by the instructor. The social graph built from student interactions on the social media tools was analysed at several levels of granularity

(whole graph, communities and individual perspective). While we acknowledge that our framework is not exhaustive, we believe that the selected methods could prove useful in similar social learning scenarios.

Next we experimentally validated the proposed conceptual knowledge extraction framework on *eMUSE* platform. We were able to satisfy, to some extent, the pedagogical knowledge needs mentioned Table 3.3.1. In section 3.4.1, we managed to determine the collaboration network and quantify collaborations over several graph granularities (KN1 and KN2). By analysing various traits of the collaboration process through SNA metrics, we emphasised salient students and teams (KN3). Through detecting communities of collaboration, we were able to better comprehend the collaboration patterns (KN2-6). Our analyses showed that the course environment adequately supports both intra-team and inter-team collaboration (KN4). We argued that students' knowledge level, background in using social media and face-to-face meetings might have an impact on the collaboration patterns, as proof of the influence of external factors (KN5). We also observed the impact of the instructional scenario and teachers' instructions on promoting intra-team collaborations (KN6). Finally, by providing social graph plots and ranked lists of students, we managed to present the extracted knowledge in a clear and comprehensive way (KN7). Therefore, we successfully validated the proposed conceptual knowledge extraction framework on our *eMUSE* social learning environment. Nevertheless, we acknowledge that further research is required in order to fully satisfy instructors' knowledge needs.

As a further commitment we developed *StudentViz*, through which an average computer literate instructor is able to observe the collaboration status on different levels of network granularity, to emphasise different traits of the collaboration spectrum through various SNA metrics, and to turn their focus towards specific learners / teams; also, studies can be conducted by filtering the collaborations by media source and/or time-frame. Hence, we consider that all basic visualisation needs are successfully addressed, and an adequate support for gaining valuable insight is provided. Overall, the main advantage of *StudentViz* compared with generic network visualisation platforms (as surveyed in section 3.5.1) is its ability to provide simple and pedagogically relevant SNA visualisations, while being easy to use by the average teacher. This is in line with the principles behind similar

tools [63, 184], but applied in novel educational settings; students' collaboration is supported by social media tools (interaction data being retrieved from *eMUSE* social learning environment), rather than by classic discussion forums (data being retrieved from traditional learning management systems).

As future work, we plan to extend *StudentViz* with more visualisations, such as three-dimensional plotting methods, time-based approaches and additional representations, like the one depicted in Table 3.5.1. Another desirable functionality is the visualisation of two graphs side by side, with the nodes having a mirrored position, to provide an easier comparison. A graphical selection tool could also be developed, that will allow teachers to extract sub-graphs for further analysis. The Data view may be extended with advanced filtering methods, that will give instructors more options to focus on particular parts and traits of the graphs. A module where teachers can easily define their personalised metrics may also be included. Adding a content analysis dimension to *StudentViz*, as in [51] or [184], would be an interesting research direction. As well as, using the textual analysis methods presented in section 3.2 to extent the weighting methods presented in section 3.5.2. This could provide further information to the instructor, that could be used to offer personalised feedback and interventions to the students. Finally, *StudentViz* could be used to explore data from other social learning environments; the DtoG module is flexible and could easily be extended to accommodate different data sources. Consequently, our goal is to perform also more experimental studies, in different settings and with a larger number of students and teachers; collecting and analysing instructors' subjective experience and opinions on *StudentViz* would provide additional validation for our platform. Furthermore, investigating students' collaboration patterns over several semesters and courses could provide a valuable insight into the social learning process.

There is a central difference between the old and the new economies: the old industrial economy was driven by economics of scale, the new information economy is driven by economics of networks.

Carl Shapiro

4

Organisational Network Analysis

Organisational Network Analysis (ONA) is a branch of Network Science/Complex Networks Analysis, devoted to analysing formal and especially informal networks that occur in an organisational context [166]. Two types of networks could be found in any organisation, formal and informal networks. Formal networks are well defined and regulated by the organisation, e.g. the organisational chart. Informal networks are self organising and hard to perceive. Diverse informal networks can be found in every organisation, but they can be categorised as: social or working networks; For example a social network may be constructed based on *who eats with whom* during lunch. Such networks can help in personnel relocation, in a manner that will augment the social environment of the organisation. Working networks try to capture how work is really done, e.g. a network of *whom to ask for advice*.As reported by Arent Greve, both human capital (the knowledge we have) and social capital have positive effects on productivity, but surprisingly social capital is the dominant effect [100]. The social capital is defined as the ability to connect to other people that give: access to information, access to resources

and career sponsorship [198]. Thus besides reinforcing information flow, a good organisational network also enhances productivity.

In recent years several studies have been carried regarding the effectiveness of information exchange in different types of organisations [58]. It is argued that good communication within an organisation has several positive effects on its productivity. In this section we use the term good communication to refer to the amount of information that can be exchanged between any two given individuals.

Good local interactions create a global outcome that no one could accomplish alone [129]. Well formed communities can sustain a good idea through the paces needed to put it in practice [57]. Positional silos, one of the motives that can reduce trust and communication, are not present in a healthy communication environment [166].

Paper [128] presents the generally accepted mathematical proof that the amount of knowledge reaching the addressee is directly dependent on the length of the path between the source and the target. In other words the more intermediaries are present between two persons the less amount of information is exchanged.

Important multinational companies, like *Halliburton*[1] having over 70000 employees in 80 countries, use ONA to enhance the work environment [58]. ONA was employed due to the fact that the sheer size of the company made it very hard for employees to exchange knowledge over traditional means. After implementing ONA techniques, as part of daily routine, the company's stakeholders discovered that the company possessed far more knowledge than anticipated before. Thus productivity grew together with the amount of revenue.

The analysis of business processes in organisations can be focused on several aspects including qualitative and quantitative features. Qualitative features usually refer to formal verification of correctness properties like safety and liveness [10] , while quantitative aspects can address performance evaluation using simulation or interaction complexity based on complex networks analysis – i.e., the focus of this chapter.

[1] https://www.halliburton.com/en-US/default.html

4.1 Background & Related Work

4.1.1 Ontologies

Due to the vast applicability of CNA in diverse domains of research there can be widely varying viewpoints and assumptions towards phenomena that are basically the same. As people interested in CNA might have different research backgrounds and the network models can be built from different perspectives, different sets of overlapping jargons may be used for describing the resulting models. This leads to the lack of shared understanding of the resulting models, thus hindering: communication, inter-operability, and re-usability [218].

The need for a unifying framework for organising and describing the knowledge in the field arises in this context. Ontologies have been used successfully in diverse cases: biology [7], public transport [225], medicine [136], linguistics [113]; to define a model that makes explicit the basic conceptualisation of a knowledge domain. T. R. Gruber which is one of the most known experts in creating ontologies defines the term ontology as a *specification of a conceptualization* [102]. According to Guarino [103], an ontology "is a logical theory accounting for the intended meaning of a formal vocabulary". The building blocks of an ontology are the concepts and the relations among them. The definitions, properties and constraints of concepts are defined as logical axioms.

Thus, we argue an ontology model for capturing the knowledge domain of the types of complex networks is well suited to our needs. Moreover, ontology creation is a well established practice in the academic world. Finally, the capabilities of consistency checking and automated classification of ontology reasoning endow our approach with increased reliability [152].

4.1.2 ONA Phenomena & Measuremes

In order to understand results presented in this chapter we have to take a quick overview of some of the phenomena/measurements that Organisational Networks Analysis addresses/uses. Please remember that when we refer to a path in this chapter we mean the shortest path possible between two nodes. As we can not hope to describe all metrics in this work, we shall present those that are most rel-

evant to our work, for further information we refer the reader to papers [34] and [68].

Stanley Milgram [155] and Watts et al. [227] discovered and defined the *Small World phenomenon*. Otherwise called *six degrees of separation*, this phenomenon is found in many real world large networks, where contrary to the size of the network the average path length between two nodes has a very low value (6 or less). Barabasi et al. [17] showed that real world networks have a *scale free degree distribution*, also called *Pareto* or *Zipf* distribution. This means that very few nodes have high *Degree* while the majority has almost the same very low *Degree*. An explanation for the appearance of the *scale free distribution of degree* is the *preferential attachment* phenomenon [163] of nodes, a node has a greater probability to be linked with nodes that have high *Degree* than with nodes with low *Degree*. Another phenomenon that is of great interest for NS is *Homophily*, described as the tendency of individuals (nodes in our case) to associate and bond with similar others [150].

We propose a 2-layer division of ONA's measurements. Each layer employs its specific measurements or otherwise called metrics. The *Top layer* is concerned with the description of the entire graph. The *Individual layer* targets each node's properties. Using this layered approach, ensures that the network is analysed from different abstraction points.

Top layer

The following metrics have the purpose of creating an overview of the entire network. Individual node values may differ largely. It is important to corroborate general metrics with those in the *Individual layer*.

The *Diameter* is the longest path in the graph. The *Radius* represents half of the *Diameter*. When looking at this measures we can figure out how deep a network is. When paring these metrics with the number of nodes we can see if the network backs information transaction. A network with a small number of nodes but large *Diameter* would imply a poor communication. The smaller the value the more the transfer of information is supported by the network.

The *Average path length* is computed by adding all path lengths within a graph

and dividing them by the total number of paths. Remember that we are talking about shortest path lengths. This metric is an important indicator of how fast the information spreads within a network. The smaller the value of this metric the better. In the *Small World Phenomenon* [215] the *Average path length* is equal to about 6, but remember that this applies to very large networks with billions of nodes. In paper [128], Noah Friedkin has proven that a network with a larger than 3 *Average path length* does not support information exchanges as the knowledge transmitted is very fuzzy. So in our case we should be looking at values smaller than 3.

The *Erdös number* describes the *collaborative distance* between a person and mathematician Paul Erdös, as measured by authorship of mathematical papers [101]. Used in the context of a graph, a node's *Erdös number* represents the length of the path between that node and a predefined starting node. For example if node A is directly connected with node B, and A is considered as the starting point. Then A has an *Erdös number* of 0 and B an *Erdös number* of 1. This measure is good for figuring out how further away are the rest of the nodes from the starting node, which is typically chosen as the most central/involved person. The average *Erdös number* is just the sum of all nodes' *Erdös number* over the number of nodes. This points out how well is a person connected with the rest of the network. Or how fast (in average) information travels from the starting node to the rest of the network. Thus, the smaller the values of these metrics the better the information exchange in the network.

The *Clustering Coefficient* is a measure of how much nodes cluster together, on a scale between 0 and 1 [227]. If we think that a node is a person we can explain this measure as follows. If two people have a common friend there is a significant probability for them to meet or to become friends, due to a phenomenon called *triadic closure*. Otherwise said, how many pairs of our friends are also friends among them. The amount of *triadic closures* is depicted by this measure. In the case of a complete graph the value is equal to 1. This measure can be interpreted of how much does a network support collaboration between individuals, the higher the value the better the network is with respect yo this criterion. The *Average Clustering Coefficient* is calculated by summing all nodes *Clustering Coefficient* and dividing by the number of nodes [134].

Modularity was designed to measure the strength of division of a network into

modules (also called groups, clusters or communities). *Modularity* is defined as the number of edges falling within groups minus the expected number in a equivalent network with edges placed at random [165]. The possible values of *Modularity* lie in the range $[-1/2, 1)$. It is positive if the number of edges within groups exceeds the number expected on the basis of chance. To determine the modularity we used the method presented in [32]. This method is also used to determine inner communities of networks.

Individual Layer

The subsequent measures help describe a node's/individual's properties in the context of the network.

The *Closeness* centrality measure of a node is defined as the average shortest path from that node to every other node. Let there be N nodes in a graph, and we pick one node j. First we calculate the shortest paths from node j to the other N-1 nodes. Then we sum the values of these paths and divide the sum by N-1, and we get the *Closeness* value for node j. It can be interpreted as a measure of how long it will take for information to spread from one node to other nodes sequentially, or how long it takes for information that raises in the suburbs of the network to get to a certain node. It is especially useful to asses important people in the company, like managers.

A node with high *Betweenness* appears more often in paths between nodes in the network. The higher the metric of *Betweenness* then the more important that node is for the transaction of information. In the shortage of a node with high *Betweenness* the exchange of information would take a much longer path, or it would not be possible at all. In the case of a complete graph *Betweenness* is equal to zero as there is no need for intermediaries. Generally speaking is advisable not to have few nodes with high *Betweenness* as the network becomes over resilient on them. When a node has a *Betweenness* value which is significantly greater than the others, it becomes overwhelmed with data, thus increasing the chances of information loss.

You can establish the *Betweenness* of a node, just following the next 3 steps:

1. set a counter for each node equal to 0, this counter will count how many

shortest paths pass through that respective node.

2. take all shortest path that pass through that node, for every paths add 1 to the counter above mentioned.

3. divide the counter value with the number of all shortest paths in the graph, and you get the value of *Betweenness* for that specific node.

PageRank is probably one of the most known algorithms for determining a node importance, as it was used by Google in its search engine [39]. Let's say that we have a node called A and we want to calculate its *PageRank*. This algorithm takes in consideration both the number of links that point to the node A, and their importance as well. To be recognised as an important node you have to be linked to other important nodes. Simply having a high number of links is not enough if they are not good quality links. Again it is advisable that there should not be just few nodes with high *PageRank*, to avoid high dependency on them.

4.1.3 Organisational Development Methods

Agile development methods were created as a response to problems encountered with classical rigid methods of developing a product [64]. For example, by using the classical waterfall methods all issues that may appear had to be envisioned and taken in consideration at the beginning of the project [33], which in the best cases is a daunting task event with simple products development. When considering the ever growing complexity of products in our era this becomes almost an impossible job. Also during the project development there is no easy way to implement changes, in the best case the changes need to be postponed for a future iteration of the product. Bright ideas and/or stakeholder's changing demands, that usually appear during the process of development, have little chances to be implemented. Several other problems were detected with the use of waterfall models, like: hard to estimate the duration of development, difficult to capture all of the future user's preferences from the beginning, large overhead with organising teams, etc.;

Some of the issues are addressed by using iterative and incremental development phases, where requirements and solutions evolve through periodical exchange of feedback from future users and Stakeholders. By using self organising

cross-functional teams the overhead on managing people is dramatically lowered. We enumerate some other concepts that are promoted by agile development methods like: adaptive planning, evolutionary development and delivery, time-boxed iterative approach; All these concepts together create a healthy environment for the development of good products/software.

In this chapter we shall focus on one single agile development method, called SCRUM. We motivate this choice as SCRUM is one of the first created agile development method, and one of the most used one, if not the most used. Companies like IBM[2] and organisations like the United Nations[3] use it with success.

SCRUM

An article from January 1986 published in Harvard Business Review [210] is considered as being the starting point of SCRUM. The article proposed new successful methods of product development that arose in the United States and Japan. These were developed with the purpose of adapting to a "fast-paced, fiercely competitive world of commercial new product development", where "speed and flexibility are essential". Later DeGrace and Stahl [65] analysed the game of Rugby, and noted that self-organising teams can be successful towards reaching their goal. These efforts resulted in a framework for developing products in time-boxed cycles of work, formalised by Jeff Sutherland and Ken Schwaber [208].

SCRUM defines 4 roles: the Product Owner, the Team, the Scrum Master, and the Stakeholders. The Product Owner is responsible for identifying product features and prioritising them into a list, called Product Backlog. The list is continuously refined and adapted to match the pace of the development. Also the list may be updated at any time with new requirements as the project is being developed. By carefully defining the elements in the list, the Product Owner maximises the return of investment.

The building block of any SCRUM process is the Sprint. This a time boxed period of time, usually from 2 to 4 weeks long. The product is built during a series of iterative Sprints, with each Sprint being finished the product gets closer to completion. Before each Sprint starts, the developing team chooses from the Product

[2] https://www.ibm.com/us-en/
[3] https://www.un.org/

Backlog the items that will be functional by the end of that Sprint.

The Team is responsible for creating the product defined by the Product Owner. By creating cross-functional teams that include all the knowledge and expertise needed, the team becomes highly autonomous. The team is self organising and decides by its own what jobs are going to be developed in a fixed time frame. Teams are stable in the sense that members remain the same over long periods of time. Multitasking across different projects or teams is highly discouraged, this way higher productivity is achieved.

The Scrum Master guides and educates the team through the process of SCRUM, by making sure all the rules of SCRUM are respected. He or she keeps outside involvements from disturbing the team. No matter the kinds of impediments that arise along the road of development he/she must do whatever stays in his/her power to resolve them. He/She must not enforce tasks, but he/she must make visible all possible problems that may be encountered. The Stakeholders are those that will use the product. They communicate with everyone especially with the Product Manager. They provide the required feedback to make the product appeal to their needs.

Communication during SCRUM is done on several occasions. Before the start of every Sprint: the Team, Product Manager and Scrum Master; establish the tasks to be completed in the following time frame. This is called the Sprint Planning Meeting. Team members communicate with each other during the daily SCRUM, which is a daily meeting held among them, with the purpose to update the status of their work progress. The most important meeting concerning the exchange of information is the Sprint Review. Here all the people involved in SCRUM attend, together with anyone else interested. The goal of this meeting is to review the product developed until that moment and give valuable feedback. This meeting takes place at the end of each Sprint. The Sprint Retrospective is another meeting held immediately after the Sprint Review. Here the team members together with the Scrum Master inspect and adapt the process of development. The Product Manager may attend, but this is not mandatory. The Product Owner is free to exchange information with the Stakeholders and Scrum Master at any moment. The Scrum Master has even more liberty regarding communication exchange, as he can be approached at any moment by anyone.

Hierarchical model

A hierarchical structure follows the layout of a pyramid. Every employee in the organisation, except one, is subordinated to someone else in the organisation. Members communicate with their subordinates and with their respective superior. One advantage that may be also a disadvantage is that this structure reduces information flow.

Employees are encouraged to narrow their field of focus and become experts in one domain. Individuals become loyal to their departments and look out for the best interest in their area. Other downfalls like increased bureaucracy hinder the organisation's speed of adapting to change.

4.2 Agile Development Organisation Method

Agile software development is all about creating a healthy environment where everyone concerned with the development process collaborates efficiently. The major advantage of agile development methods is that they take into account the way the human society works and interacts. This section's purpose is to analyse one of the most used agile development methods, SCRUM, from the point of view of information flow and exchange. Having as a model a software company we shall create graphs/networks that capture the transfer of information from different perspectives. In analysing the information flow through the networks we shall use Organisational Networks Analysis.

The information exchange graphs are created with the purpose to approximate three perspectives of information flow: Stakeholders' requests, between teams skill exchange, inside teams skill exchange; We shall prove that SCRUM's effects on organisational structures has important benefits on the transfer of information. To measure these benefits we shall compare SCRUM's organisational structure with the classical hierarchical structure.

We consider two software companies that are almost identical. They have equal number of individuals, you can even imagine that they have the same individuals. The tasks and products that have to be completed are carbon copied. The only thing that distinguishes them is the way they are organised. One is based on a

hierarchical structure, while the other has the structure that results from implementing SCRUM. You can think that they are the same company, but in parallel universes with just a single distinction. We hope to get valuable conclusions by comparing how information is exchanged within each of these companies

4.2.1 Modelling the Organisational Structures

A graph (or network in NS) is composed of *nodes and links*, with each link connecting two nodes. The graph can be weighted, i.e. a label (usually a number) is associated with each link, or unweighted. Links can be unidirectional (or directed), i.e. the source and target nodes are specified, or bidirectional (or undirected). Self-looping graphs permit a node to directly connect with itself. Nodes can be seen as an abstraction of: people, entities, concepts etc. A link between two nodes models a binary relation between those nodes and has a domain-specific interpretation: physical interconnection, shared similarity, interaction, etc. In this chapter we address bidirectional, unweighted and non-self-looping graphs. They are the most common types of graphs found in the literature. The other types of graphs can be converted to this type without losing valuable information.

Before we start modelling the network structure we have to establish which are the elements that form the network. The nodes will represent either individual actors/persons or teams. Edges between nodes will represent a channel of communication among them. The most used mean of constructing informal networks is based on email logs. Email is considered to be the standard for exchanging work knowledge within an organisation. But email can also be used to socialise. Therefore an email exchange network manages to capture both working and social communication.

Modelling SCRUM in a network structure

We shall use the following notations in the graph models of SCRUM: Tn for team number *n*, PM for Project Manager, SM for Scrum Master, SK for the group of Stakeholders; In figures where teams members shall be depicted, each member receives a notation Dn, where n is a number. Every above mentioned entity is represented by a node. If two entities communicate, then a link between them shall

be present.

In order for this paper to be also of practical value, we got inspired from the world of software development. We decided that we would like to model a company branch that is able to create a software product of medium complexity. This means that there should be multiple working teams at the same time approaching distinct tasks. But we do not want too many teams as it would become harder to track the differences between the SCRUM structure and the tree-model structure.

We want to define a representative sample SCRUM network for our analysis, to support the relevance of our results. Team sizes in SCRUM may differ, although it is recommended not to be larger than 9 individuals. This way the team can manage tasks on its own, with minimum overhead in organising its members. We decided that every team should have 9 members. In this manner we shall test SCRUM at it's maximum tolerated capacity per team. Based on the rationale of the above paragraph we decided to use 3 teams, totalling 27 members. Besides these individuals our network will contain the Project Manager and the Scrum Master. Also the Stakeholders will be represented as a single node, their number may vary across the process of development, but they are not an issue of interest in this paper.

While we established the number of nodes of our sample network to be 30, we must now establish the edges interconnecting these nodes. First we will take a look at the knowledge flow. In SCRUM the flow of insights from the Stakeholders to every other participant is direct. For simplicity of presentation we decided not to represent each team member. They will be represented by their team node. You can observe the network in Figure 4.2.1.

Next we must create the graphs of expertise exchange by defining the edges between nodes. During the Sprint Planning Meeting individuals from all the teams are present, so in this case cross team communication is also direct. The change of expertise between teams during development is rarely needed in SCRUM, as each team is conceived in such a manner that all needed expertise is already present. SCRUM encourages team members to have multiple skills, this makes highly probable the case when an expertise is mastered by two or more people in the same

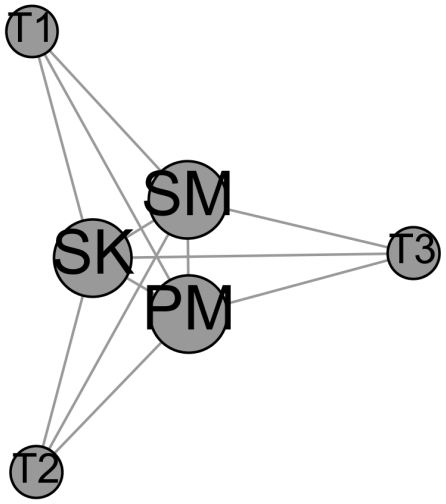


Figure 4.2.1: SCRUM network of knowledge diffusion

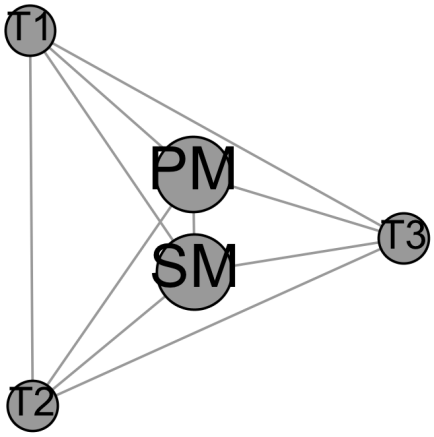


Figure 4.2.2: SCRUM network of skill exchange

team. Thus you can imagine that the need for additional expertise from outside the team diminishes even more.

However we want to consider the worst case scenario from the point of view of skill exchange. This happens when a Sprint is in progress and there is need for some expertise from another team. Then the Scrum Master will facilitate a meeting between the two teams, he will not act as a middleman in the exchange at any time.

So again we have direct transfer of skills. See Figure 4.2.2 for the network. As the Scrum team is self organising and of relatively small size, direct communication is possible between everyone. In this case the network of expertise transaction is identical with a complete graph with 9 nodes.

Modelling the hierarchical network structure

The following notations will be used in the network models of the hierarchical model: Project Manager (PM), Business Analyst(BA), Software Developer Manager (SDM), Quality Assurance Manager (QAM), Stakeholders (SK), Programming Managers (Prm), Programmers (P), Design Manager (Dm), Designers (D), Quality Assurance employees (QA), Team of Designer (Ds), Team of programmers (PTn);

In a traditional hierarchical scheme, there are several compartments, each one commissioned with a specific function. To better illustrate these compartments we took as example a software company, like we did with SCRUM. The following departments can be usually found: business analysis, software development, quality analysis and management; The number of nodes that are going to be present in this model is 30, equal to the one in the SCRUM model.

We start describing our tree structure from the top of the pyramid, position held by Project Manager. Going one level down we can find the following individuals: Business Analyst (BA), Software Developer Manager (SDM) and Quality Assurance Manager (QAM). Under the QAM we can find the 3 other quality assurance subordinates.

The branch with the most numerous personnel is the Software Department, it has 24 nodes. This department is organised in the following way:

- at the top we can find the Software Developer Manager (SDM)
- on the second level there are 4 middle level managers: 3 Programming Managers (Prm) and 1 Design Manager (Dm).
- each middle manager supervises his own branch. Each Prm has 5 programmers (P) that form a programming team (PT), and the Dm has 3 designers (D).

You may notice that there are multiple divisions of size 3: 3 QAs, 3 Prm, 3 designers. This was envisioned with the purpose of a relevant comparison with SCRUM, which has 3 working teams. If we consider that each Prm has 5 programmers and we add a QA and a designer, we get that a total of 8 people may work together on the same set of tasks

When creating the network of knowledge dissipation we assumed the following :

- the Project Manager together with the BA, QAM, SDM meet directly with the Stakeholders.

- although the Design manager is not a top manager in our configuration, he/she needs to meet with the Stakeholders directly. The interface of a software product is highly dependent on what Stakeholders wish to see. Usually the SDM has a background as programming fellow, thus he might not be qualified to sketch the interface.

- Programming managers might interact with Stakeholders, but this does not occur on a regular established timetable. Usually the software requirements are captured by the SDM and then allocated to its Programming managers. Each Prm breaks his allocated assignments into smaller tasks and designates them to Programmers. A Prm has little to say towards the design of the entire application, he might make proposals, but the SDM takes the final decision. Good ideas based on insights might not make their way into the design. The above being said we conclude that even in the case when SK communicate directly with Prm, this can not be compared with the exchange of knowledge that takes place in SCRUM between team members and SK. Thus there will not be a link between the actors in case.

We aggregated each team of programmers into a single node (PTn) to simplify the presentation. We did the same with the designers with the difference that they were aggregated into a designers nodes (Ds). You can see the network in Figure 4.2.3.

Before creating the link between nodes we made the following assumption: individuals of equal hierarchy in the same branch communicate directly. This has

the following effects on the exchange of skills between teams:

- the Business Analyst, Software Development Manager, Quality Assurance Manager may communicate directly.
- the Programming Managers and Design Manager also exchange skills without any intermediaries.
- a programmer can not communicate directly with all the other programmers, but just with those that have the same Prm as he does.

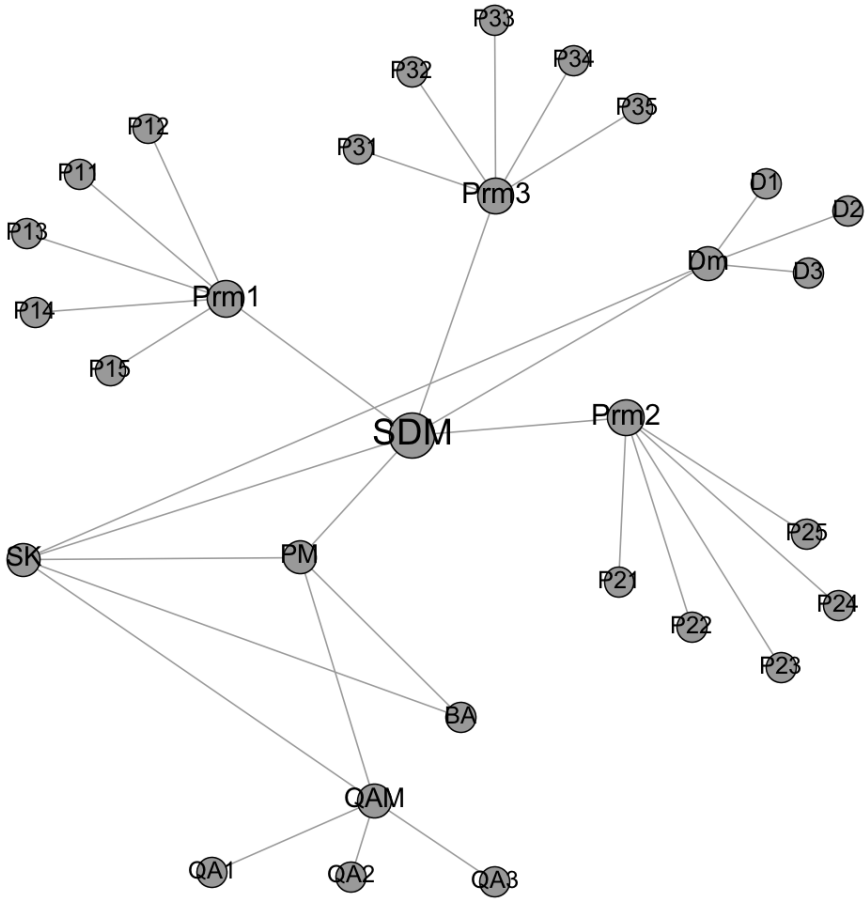


Figure 4.2.3: Hierarchical network of knowledge diffusion

In Figure 4.2.4 you can observe the network of skill transfer between teams. The leaf nodes representing programmers, designers, and quality assurance staff have been aggregated into a single node for the purpose of simpler presentation.

Next we looked upon the exchange of information inside teams. Based on the prior assumption that individuals of equal hierarchy in the same branch communicate directly, we first concluded that all team members communicate directly. But consider the fact that a programming team can not be autonomous in the process of development. A product is also developed by the designers and tested by the QA personnel. Not to mention that the product must be permanently supervised by the Prm, Dm and QAM. Even more the SDM must be aware and approve all the major points. So not all the communication is direct.

If we define the team in the sense that SCRUM does, then in the case of the hierarchical structure the team is the entire network. The probability that it might be necessary to exchange skill information between different programming teams is high. This is because the hierarchical model promotes super specialised people in a narrow domain. An organisation with just a SCRUM team can create a reliable product, but an organisation that is based on a hierarchical model can not deliver the same quality if one of its major components (e.g. the QA department) is removed. We conclude that the network of skill exchange inside teams is the same as the network of skill exchange between teams.

4.2.2 Discussions & Results

This section is dedicated to the presentation of the results of network comparisons between the two organisational models' network structure, SCRUM and hierarchical.

Knowledge dissipation

We start by comparing the knowledge dissipation from the Stakeholders to the rest of individuals. First we look at some general graph metrics, see Table 4.2.1. The *Average path length* in SCRUM is significantly shorter, so insights get quicker to team members than in the hierarchical model. The *Diameter* and *Radius* of the

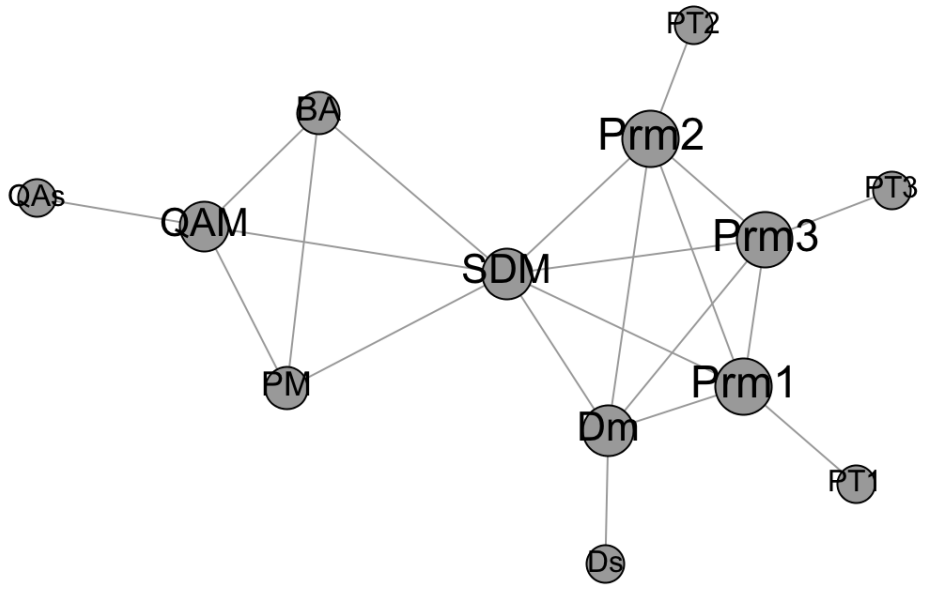


Figure 4.2.4: Hierarchical network of skill exchange

tree-like model show us that this model has more layers of authority than SCRUM, which again proves that knowledge travels slower in the hierarchical model.

When looking upon the average *Erdös number* in the tree-like model , with the SK node as starting point, we observe that in general individuals do not get to meet with the Stakeholders, as they are more than 1 step away. In SCRUM everybody has repeated opportunities to talk with the Stakeholders.

Now let us take a look at Stakeholders' metrics in both networks, see Table 4.2.2. The *Degree* metric of SK depicts how many individuals meet directly with the Stakeholders. Again the hierarchical model is out beaten by SCRUM, but this

Table 4.2.1: General graph description of knowledge dissipation.

Metric	SCRUM value	Hierarchical value
average path length	1.8	3.2
Diameter	2	5
Radius	1	3
average Erdös number	1	2.34

Table 4.2.2: Stakeholders node properties in knowledge dissipation.

	Closeness	Degree	Betweenness
SK model	2.34	5	71.5
SK SCRUM	1	29	117

was expected. The value of *Closeness* centrality tells us that in the tree-like structure knowledge travels through more people before getting to the leaves of the graph. So reliable information has higher chance to be lost in the tree-like structure than in SCRUM.

Using the *PageRank* algorithm and the *Erdös number* we wanted to see which positions in the graph are the most important for the flow of knowledge. We created Table 4.2.3 for SCRUM and Table 4.2.4 for the hierarchical model. Note that in SCRUM the most important people are the Stakeholders, Project Manager and Scrum Master, which is expected as they are the most connected people.

In the tree-like structure the most important position is that of Programming Manager (Prm). When we take in consideration that there are 3 of them, we understand how highly dependent is the knowledge transaction on them. But their *Erdös number* is 2, which says that they do not directly meet with the Stakeholders. So the most important individuals for the dissipation of knowledge from the Stakeholders do not directly have contact with them. This again is a sign that valuable knowledge may be lost or altered.

Table 4.2.3: Nodes properties in knowledge dissipation, SCRUM network.

Position	PageRank	Erdös number
SK	0.16	0
PM	0.16	1
PM	0.16	1
team member	0.01	1

Table 4.2.4: Nodes properties in knowledge dissipation, hierarchical network.

Position	PageRank	Erdös number
SK	0.04	0
PM	0.04	1
BA	0.04	1
QAM	0.07	1
SDM	0.08	1
QA	0.01	2
Prm	0.10	2
Dm	0.08	2
Designer	0.01	2
Programmer	0.01	3

Communication of skills between teams

The big advantages of SCRUM come to surface when we speak of skills exchange. All the metrics of the SCRUM model reach the ideal values when looking from the perspective of information exchange. So in this subsection we shall take more focus on the metrics of the hierarchical model.

In Table 4.2.5 we present the overall metrics of the networks. The *Diameter* of the tree-like model is 4 times larger than SCRUM's. Among the paths that reach the value of the *Diameter* there is probably no effective transfer of skills. The average path length is a little bit larger than 2, which might make us think that there is the possibility for some valuable exchange of skills.

When we take a look of how many actors in the networks collaborate directly, again the hierarchical model falls behind. In average an actor has the possibility to work directly with less than half of the other actors in the tree-like structure, see the clustering coefficient. In SCRUM everybody can work directly with every other

Table 4.2.5: General graph description of skill transfer between teams.

Metric	SCRUM value	Hierarchical value
Diameter	1	4
Radius	1	2
average path length	1	2.11
clustering coefficient	1	0.41

Table 4.2.6: Nodes properties in skill transfer, SCRUM network.

Position	PageRank	Closeness	Betweenness
PM	0.2	1	0
SM	0.2	1	0
team member	0.2	1	0

actor.

In SCRUM all the actors have equal importance. When *Betweenness* is equal to 0, this means that there are no intermediaries between two points. So the possibility of loss of information is highly diminished. See Table 4.2.6 for nodes' properties in the context of the SCRUM network.

Upon analysing the hierarchical structure, see Table 4.2.7 for metrics' values, we noticed the following:

- managers tend to be relatively close to everyone in the network with a *Closeness* metric that is about 2. So at the level management there is good chance of skill exchange.
- the SDM, QAM, Prm and Dm are responsible for almost all the exchange of skills in the entire network. The SDM has to supervise as much skill exchange as all the other above mentioned actors. So the network is highly dependent on them. Also the amount of information that they must manage is overwhelming, at least in the case of the SDM.
- programmers and designers have the lowest chance to reliably exchange skills among them. They have the highest *Closeness* centrality in the entire network.

Table 4.2.7: Nodes properties in skill transfer, hierarchical network.

Position	PageRank	Closeness	Betweenness
PM	0.07	2.08	0
SDM	0.15	1.41	32
BA	0.07	2.08	0
QAM	0.09	2	11
QAs	0.03	2.91	0
Prm	0.11	1.66	11
Dm	0.11	1.66	11
Ds	0.03	2.53	0
PT	0.03	2.58	0

Table 4.2.8: Distance from the programming teams to the QAs and Ds.

	QAs	Ds
path length	4	3

Having found that in average designers, programmers and QAs are less likely to exchange skills, we analysed the path lengths between them, see Table 4.2.8. With such elongated path length the probability of skill exchange is very low. When taking in consideration that these individuals represent 70% of the organisation's personnel, then we conclude that the hierarchical model does not promote skill transaction.

Skill exchange inside teams

When talking about inside the team skill exchange in SCRUM, apart from the fact that the network becomes a complete graph with 9 nodes, the metrics' values remain the same as in the case of between teams skill exchange. As argued before in the case of the hierarchical model, the network is the same as the one in the previous subsection. This being said we decided there is no further need for comparisons between metrics. The result would be the same as above.

Additional remarks

In the case of SCRUM the networks depicted in this paper support the core-periphery theory [129]. The core is composed of some highly connected nodes, in our case:

PM, SM, SK. This core supports the exchange of information, and creates good knowledge of what is happening in the organisation. The periphery nodes can reach easily the core, but not so easily the other nodes in the periphery, in our case team members. Diversity is backed up this way, which is necessary for creation on new and unique ideas. These unique ideas are essential for well accommodation of an organisation to new contexts.

4.3 Types of Complex Networks

This section presents a new ontology that enables the knowledge-based analysis of complex networks. The purpose of our research was to develop a new approach for the knowledge-based analysis of complex networks based on various network attributes and metrics. Our approach is both easy to use and easy to understand by a human. It facilitates the automated classification of different types of networks. For the creation of this ontology we applied an already known methodology from the scientific literature. The ontology was also enriched with our own developed methods. We applied our ontology to the analysis scenarios of complex networks obtained from real world problems, thus supporting its generality, as well as its usability across domains.

Our current understanding of the surrounding environment, either geographical or biological, shows us that nature is formed out of complex interconnecting systems. Networks created by these systems support phenomena that are far from being deterministic trough traditional methods. Each element influences the network, while the network puts its mark on every element. Now we can say with certainty that the *butterfly effect* imagined by Edward Lorenz is truly possible.

NS defines some basic types of graphs by trying to understand and model the phenomena that led to their creation. Knowing the type of a complex network is very important as it gives very powerful insights into the model or phenomenon that it represents. For example an organisation modelled as a *Watts-Strogatz* [227] type of complex network supports very well the exchange of information, as it incorporates the *Small World* phenomena [155]. But unfortunately this type of graph does not support the phenomenon of *Homophily* [150], that is also found in many real world networks.

As far as our knowledge expands, we are not aware of any other work that defines an automated process of determining the type of a complex network based on its attributes. Moreover, our proposed knowledge-based approach has the following features:

- It is easy to use, understand and share by humans.
- It is able to propose new classes of complex networks, in addition to those that can be found in the research literature.
- It is reusable and expandable, i.e. it captures in a reusable way existing classes of networks, as well as it allows the addition of new classes as the knowledge in the field will expand.

The analysis techniques considered in our approach are inspired by real world applications. We are able to exemplify the applicability of our knowledge-based method to real world problems from different application domains.

4.3.1 Creating the Ontology

Overview

As far as we know, this is the first attempt to build a *Complex Networks* ontology, Moreover, we could not find any other ontologies that are somehow related with our purpose. Therefore, we had to start from scratch and create a new ontology. We used the following steps to define our ontology [168]:

1. Determine the domain and scope of the ontology.
2. Enumerate important terms in the ontology.
3. Define the classes and their relations to each other (hierarchy).
4. Define the properties of classes.
5. Define restrictions on the properties.
6. Create instances.

Vision on the use of the ontology

Our vision is to use the ontology in various application scenarios of CNA. For a given problem, CNA starts by formulating a set of specific questions regarding the network under analysis. Usually such question can be answered by performing a thorough analysis of the network using the available CNA computational tools[4]. The analysis returns a relevant set of metrics and their respective values. Based on those metrics, the network can be classified into a certain known class. This classification enables the user to obtain additional insight on the properties of the underlying network.

The ontology should be able to classify networks in different categories, depending on their support of information exchange. The basic competency questions that arise are:

- What type of graphs are studied with this ontology? Graphs can be: directed/undirected, self-looping/non-self looping or weighted/unweighted;
- Which are the types of complex networks (complenets)? Which of them are relevant and widespread enough to be introduced in the ontology?
- Based on the determined types of complenets, can one of them be a subtype of another type? Can a complenet be at the same time of two or more types?
- Which are the canonical (present in the scientific literature) characteristics that define each complenet type?
- How can we quantify complenets' characteristics? Metrics are used with this purpose, but could we include phenomena or other methods of expressing those characteristics?
- Which are the metrics and phenomena that can express the characteristics of a network? Considering the fact that different metrics and phenomena may determine the same characteristics, which will be used in the ontology and which will be considered to bring redundant information?

[4]Like for example Gephi: https://gephi.org/

- As CNA is a 3 layer stratified analysis, each layer having its specific metrics, how will the metrics be represented in the ontology?
- Besides canonical ways of determining the type of a complenet, are there any other reliable ways to determine the type of a complenet? Which are the types of complenets that can be determined with alternative and reliable methods? How can these methods be used to enrich the ontology?
- Due to the fact that exemplification of the ontology's power is imperious, can we introduce powerful examples in the ontology? Well known and used networks could be used with this purpose, but which should we chose?

For the purpose of explaining our vision we shall imagine an application domain in which links between nodes model the exchange of information. Our purpose is to find insights on how the information spreads in the network. In this particular case, the following questions should be answered after the type of the complenet was determined:

- Has information good chances to spread in the entire network?
- How much time (estimated in the number of steps) it takes until the information has spread in the vast majority of the network?
- Can we compare two different networks and answer which supports better the information exchange?
- If information can suffer from alteration at each intermediary step, which networks reduce information alteration?

The proposed ontology will help us to easily classify new complex networks based on their metrics, while the resulting classification will be easy to understand and use. For all the questions iterated above you will find the answers in the next section as we shall describe the ontology.

4.3.2 Defining the Ontology

Our ontology is expressed using OWL 2 Web Ontology and it was created with the help of Protégé [111] open source software. The ontology contains: classes, object properties, data properties and individuals. NS metrics shall be presented while explaining the ontology construction. The ontology is available via a public *GitHub* repository[5].

Classes

OWL classes represent concepts and are interpreted as sets of individuals. For example, analysing the class hierarchy from Figure 4.3.1, note that the top level class is called *Thing* and it has 4 sub-classes (we shall call them *Top-level Classes*) as follows:

General Graph metrics is a *Top-level Class* that represents value partitions of multiple graph metrics. These metrics are: *Clustering Coefficient* [194], *Density* and *Modularity*; As all these metrics follow almost the same schema we shall only present the *Modularity* metric. *Modularity* was designed to measure the strength of group divisions (also called groups, clusters or communities) in a network [131]. The possible values of *Modularity* lie in the range $[-1/2, 1)$. It is positive if the number of edges within groups exceeds the number expected on the basis of chance. *Modularity's* values are represented by the following disjoint sub-classes: *Negative Modularity* and *Positive Modularity*; Also *Positive Modularity* contains sub-classes that divide the $(0, 1)$ range of possible *Modularity* values in 5 equal partitions: *VeryLowGraphModularity* $(0, 0.2]$, *LowGraphModularity* $(0.2, 0.4]$, *MediumGraphModularity* $(0.4, 0.6]$, *HighGraphModularity* $(0.6, 0.8]$ and *VeryHighGraphModularity* $(0.8, 1)$; The sub-classes of the *Clustering Coefficient* and *Density* metrics follow the same schema as the sub-classes of the *Positive Modularity*.

Graph Structure is a *Top-level Class* that contains elements which define the overall structure of a complex network. These elements are represented by the following subclasses: *Betweenness Concentration*, *Distribution of Degree*, *Number of Connected Components* and *Structure Type*;

Betweenness Concentration is based on the *Betweenness* metric. A node with

[5] Github repository: `becheru.github.io/Ontology`

high *Betweenness* appears more often in paths (shortest paths) between nodes in the network [34]. If we normalize *Betweenness* such that each node has a value between 0 and 1 with the total *Betweenness* being 1, we can determine which is the value of the highest *Betweenness* node of the graph. *Betweenness Concentration* has 3 disjoint sub-classes that represent the possible value it can take. If a node has a *Betweenness* value of more than 0.6, then we should chose the *Single Node Betweenness Concentration* subclass. If this is the case, then it is a sign that the graph might have a *Single Hub* structure type. If the above case is not true but the sum of multiple nodes *Betweenness* surpasses the threshold of 0.6, then the right subclass to chose is *Multiple Node Betweenness Concentration*. This is a sign of a presence of a *Multiple Hub* graph structure type. If the *Betweenness* value of every node is equal to zero than the right subclass to chose is *No Node Betweenness Concentration*.

The *Distribution of Degree* is obtained by making a histogram of all nodes' *Degree*. A node's *Degree* is equal to the number of edges that connect with it (undirected graph). Three possible distributions are present in our ontology, each is represented by a disjoint subclass: *Uniform Distribution*, *Poisson Distribution* and *Exponential Distribution*. The *Exponential Distribution* contains a subclass depicting the *Pareto (Zipf or Power Law)* distribution [1]. The *Pareto* distribution is a skewed, heavy-tailed distribution that is usually found in real world networks.

With the *Number of Connected Components* class we try to determine if the graph is connected or if it it contains multiple subgraphs that are not connected among them. This class contains two disjoint sub-classes that represent the possible values of the class: *Multiple Weakly Connected Components* and *One Weakly Connected Component*;

The *Structure Type* is used to determine if the overall structure of the graph is of the following types: *Single Hub*, *Multiple Hub* and *Core & Periphery* [36, 110, 129]; These types of structures are depicted by disjoint sub-classes of the *Structure Type* class. See Figures 4.3.2, 4.3.3 and 4.3.4 for a graphical exemplification of these types of structures.

The class ***Graph Types*** *Top-level Class* contains the classes that do the detection of different complex networks' types (defined classes). Also here we can find several examples of complex networks, they can be found under the *Named Graphs* subclass. The types of complex networks that can be found are the following:

ScatteredFragments, Random, WattsStrogatz [227], *OverConnected, GoodExchange, Barabasi-Albert*; The *GoodExchange* type of network is based on the findings of Noah Friedkin [128], good exchange of information is possible only if the *Average Path Length* is smaller than 3. The *Barabasi-Albert* type is subdivide in other 3 subtypes: *SingleHub, MultipleHub* and *CorePeriphery*. Also in the ontology you can find alternative methods of determining the above mentioned complex networks types. They are represented by defined sub-classes that have the *Alternative* prefix, e.g. *AlternativeBarabasiAlbert*. These alternative methods are developed based on our experience in the CNA field of study.

The *Named Graphs* subclass contains real world complex networks and well known graphs in the scientific literature. For example the *Ancient World* network describes the groups of humans in the ancient world. These groups are locally (in the same geographical zone) connected, but they form a global network that contains several components that have no means of communicating among them. The Aboriginal people in Australia were cut off for thousand of years from the rest of the world. The *DoplhinSocialNetwork* represents a network of socialisation between dolphins near the coast of New Zealand [142]. The *Erdos-Renyi* graph is an example of well known graph that is present in our ontology [75].

The ***Phenomena*** *Top-level Class* describes social phenomena that could be present in graphs. Our ontology describes only two phenomena as subclasses of this class: *Homophily* and *Small World Phenomenon*;

Object properties

OWL Properties represent relationships, binary relations on individuals. Object properties are relationships between two individuals. The domain of our object properties is the *Graph* class. Our ontology consists of 3 main transitive object properties: *hasPhenomena, hasMetric* and *hasStructure*; The *hasStructure* contains other 4 functional sub-properties: *hasBetweennesConcentration, hasDistributionOfDegree, hasNumberOfConnectedComponents* and *hasStructureType*;

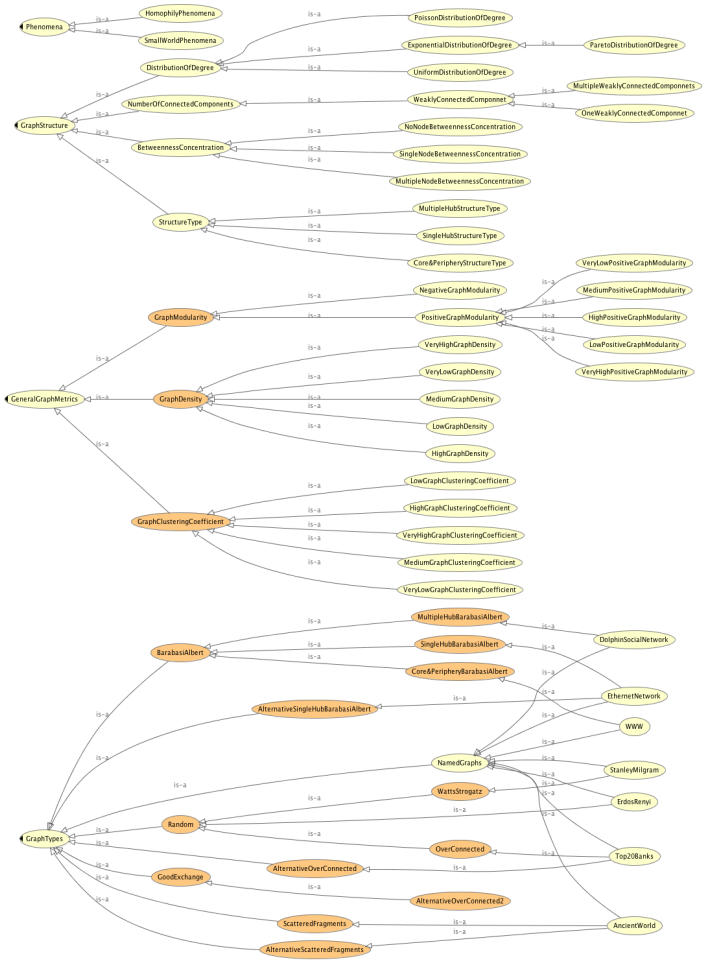


Figure 4.3.1: Figure depicting the full class diagram of the inferred model. Defined classes are marked with the orange background. Primitive classes are transparent.

Data properties

They describe relationships between an individual and data values. The ontology has two data properties, but others can be added with ease: *hasGraphAveragePathLength* and *hasGraphNumberOfNodes*. The *hasGraphAveragePathLength* depicts the length of the average path length in a graph as a number. The *hasGraphNumberOfNodes* represents the number of nodes in a network.

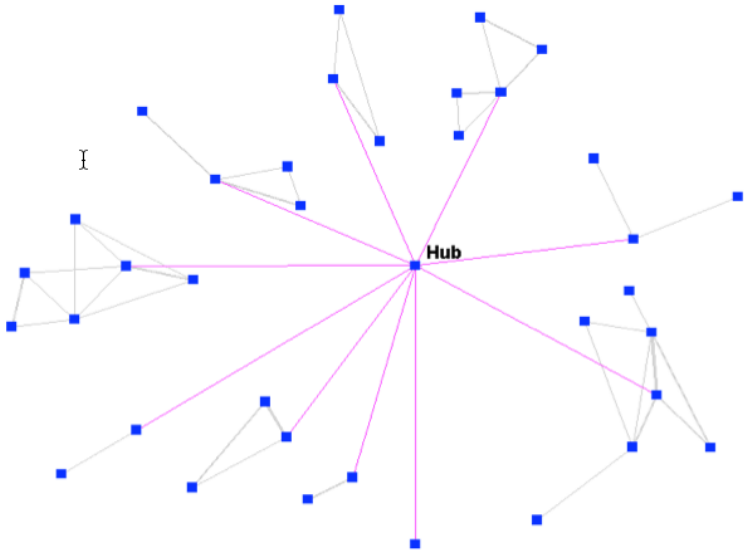


Figure 4.3.2: Figure depicting the structure of a *Single Hub* type of complex network. Figure extracted from paper [129].

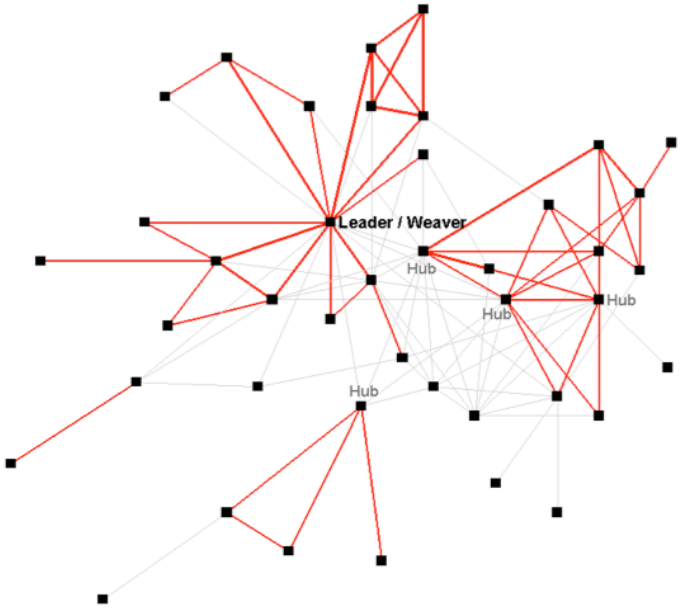


Figure 4.3.3: Figure depicting the structure of a *Multiple Hub* type of complex network. Figure extracted from paper [129].

Individuals

Individuals, represent objects in the domain in which we are interested. We have created two individuals for exemplification:

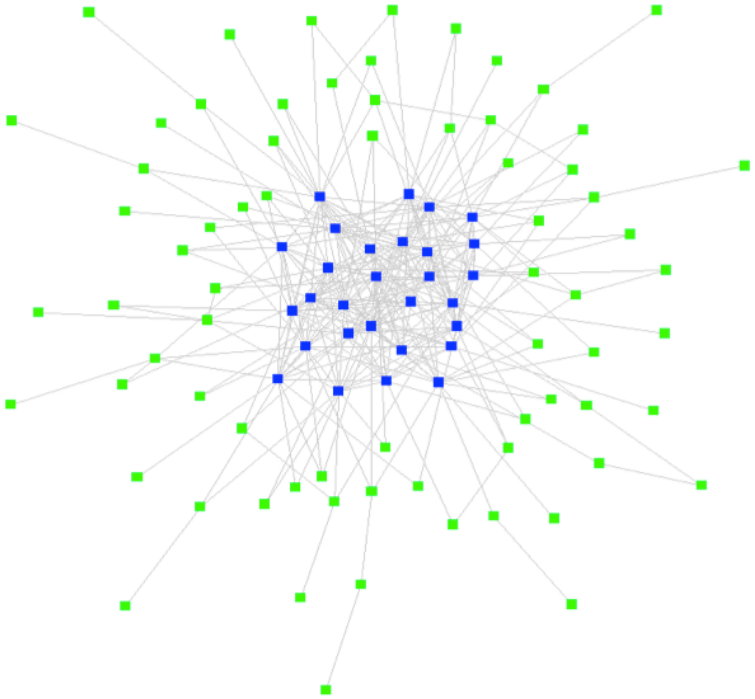


Figure 4.3.4: Figure depicting the structure of a *Core&Periphery* type of complex network. Figure extracted from paper [129].

- *ExOverConnected*: this individual is of class type *Top20Banks* with two data properties: *hasGraphNumberOfNodes*=20000 *andhasGraphAveragePathLength* with value of 1.4.

- *ExGoodExchange* : this individual has only a data type property. The *hasGraphAveragePathLength* is equal to 2.7. Also it has a *OneWeaklyConnectedComponent*, meaning that the graph is connected.

In Figure 4.3.1 you can see the inferred class hierarchy. For example the Class *DolphinSocialNetwork* is a subclass of the class *NamedGraphs*, this was given explicitly. But it is also a subclass of the class *MultipleHubBarabasiAlbert* which was inferred through our ontology. To generalise this example, every subclass of *NamedGraphs* is also a subclass of other classes determined by inference.

For the serialisation we used the *Manchester Owl syntax*, as this is easily readable. We chose to present two classes, one primitive and one defined.

The *GoodExchange* class is a defined class, thus it used to classify network types. As you can see in Listing 4.1, the necessary and sufficient conditions are under the *EquivalentTo* branch. An individual has to be a *graph* made by *one weakly connected component* and it has to have the *Average Path Length* smaller or equal to 3. The *GoodExchange* class is a child class of the class *GraphTypes*.

```
Class: <#GoodExchange>
EquivalentTo:
    <#GraphTypes>
     and (<#hasNumberOfConnectedComponnets> some
             <#OneWeaklyConnectedComponnet>)
     and (<#hasGraphAveragePathLength> some xsd:double[<= "3"])
SubClassOf:
    <#GraphTypes>
```

Listing 4.1: Example of a defined class/complex network type.

The *ErdosRenyi* primitive class has its necessary conditions, those that characterise it, under the *SubClassOf* branch. Because we present the inferred method the *Random* class appears under this branch, although we did no specify it in the asserted model. This means that the reasoner automatically detected the *ErdosRenyi* as being of *Random* type. Each object property also presents a closure axiom, i.e. those restrictions that have *only* in their definition. *Disjoint* classes with the *ErdosRenyi* class are clearly shown in Listing 4.2.

```
Class: <#ErdosRenyi>
SubClassOf:
    <#hasDistributionOfDegree> only
        <#UniformDistributionOfDegree>,
    <#hasDistributionOfDegree> some
        <#UniformDistributionOfDegree>,
    <#hasNumberOfConnectedComponnets> some
        <#OneWeaklyConnectedComponnet>,
    <#hasNumberOfConnectedComponnets> only
        <#OneWeaklyConnectedComponnet>,
    <#NamedGraphs>,
    <#Random>
```

```
13 DisjointWith:
14     <#AlternativeScatteredFragments>, <#SingleHubBarabasiAlbert>,
15     <#WWW>, <#DolphinSocialNetwork>, <#BarabasiAlbert>,
16     <#EthernetNetwork>, <#AncientWorld>,
17     <#MultipleHubBarabasiAlbert>, <#Core&PeripheryBarabasiAlbert>
```

Listing 4.2: Example of a primitive class that describes a network.

4.3.3 Ontology exemplification

In this section we present a real world application example of our proposed approach involving the complex networks' ontology. We focus on human resources in business management [21]. Note however that our proposal is general enough to be employed for a large number of other application domains. Basically, every system or phenomenon that can be modelled as a complex network can be the subject of the analysis process of our proposed ontology-based approach.

Let us imagine the following case. In a company the CEO was changed due to lack of results. The company has different branches and is present in several countries. Although the company has different branches all of them develop products for the automotive world. Some branches produce highly competitive products while others are loosing money. The new CEO is interested in bringing know-how from the profitable branches to those that are loosing them. Also he suspects that collaboration inside the company is not properly working therefor he commands a study on the exchange of information inside the company.

As email is considered the standard way to communicate important things within the organisation, an email exchange graph was created. Each employee is represented by a node, a link is present between nodes if an email was exchanged by those two nodes. With the help of CNA instruments the CEO found that following traits of the graph: *Pareto distribution of degree*, *One Weakly Connected Component* and that the structure is of type *Multiple Hub*. Next by using our ontology the CEO determined the network is a *Barabasi-Albert MultipleHub* network. By studying the properties of this network type the CEO found that there are some very few people (*Hubs*) responsible for the exchange of information within the company. Through these key employees the majority of know-how is exchanged. Also some of these *Hubs* are not even connected among them. The feedback ob-

tained is also supported by prior knowledge of the CEO, as he observed that some key people in the company rarely met. By knowing the type of network the CEO determined that ideas and know-how get lost as these key employees filter them due to the overwhelming amount of information that they have to handle.

As proven by Valdis Krebs the best type of network for balancing innovation with communication is the *Core Periphery* structure [129]. By comparing the current network type with *Barabasi Albert Core Periphery* , both present in our ontology, the CEO can determine which are the differences. Here the *Core Periphery* structure makes the difference between the two types of networks. Now the CEO knows where to focus its attention on improving the current organizational graph. For example he could schedule regular meetings with the *Hubs*. The work of Robert L. Cross et. al. could help the CEO determine what are the steps necessary to enhance the exchange of knowledge within the organisation [58].

4.4 Organisational Network Analysis in the Academic World

Based on our knowledge of previous research, we are not aware of any studies that use Organisational Network Analysis (ONA) on an academical organisation. The academic environment presents an interest to us as it is supposed to be highly dynamic. This may be a consequence of the need to stay up to date with the latest innovations. In the academic world the need for information exchange is especially relevant as academics teams are usually formed from individuals with different scientific backgrounds. This section's goal is to evaluate how well information is exchanged within a small size university, the *Romanian-German University of Sibiu*[6]. In order to better assess the organisation we retrieved data from 3 sources: email logs, organisational chart and directly from the employees. Another scope of this study is to determine which of the above mentioned sources give reliable analysis on information flow between university staff.

Generally speaking, productivity is affected by internal organisational factors and external factors. Internal factors depend on the percentage of knowledge used out of the total amount of knowledge present in the organisation. In order to use as much knowledge as possible, every piece of know-how must be easily reachable.

[6] http://roger-univ.ro/

Having a good comprehension on the environment in which the organisation has its domain of activity, helps in targeting the sub-domains where the organisation's abilities are maximised. Relying on the previous reasoning, the more the organisation is aware about its own abilities the more productive it could potentially be.

In this study information exchange is looked upon from two perspectives: formal and informal; the difference between the two perspectives is given by the source communication extraction: x is friend with y, x emails y, x eats lunch with y, etc. A network is formed from nodes and edges. A node represents an individual (person). The existence of an edge between two nodes should be interpreted as presence of communication between them. The communication considered by us is bidirectional.

Networks that arise in an organisation can be classified to be: formal and informal. Formal networks are strictly regulated and maintained by each organisation. The most obvious and most present formal network is the organisational chart, it has the role to control the way decisions and information are passed among employees. Therefore we chose to construct a network that mirrors the organisational chart. Informal networks are not administered by anyone, they form due to different habits and needs. It is necessary to construct informal networks as they are considered to capture in a more realistic way *the manner things are actually done* inside an organisation. You can consider the informal network as the backbone of any organisation, as it is hard to see from the outside but essential in the way it functions. It was suggested that even in the case when the organisational chart changes, the informal network stays the same [76], thus giving stability to the organisation. As employees exchange information through their organisation-given emails, but this varies from case to case. Thus we decided to use the email exchange graph as a source for our informal networks. This study should also establish if information exchange through electronic mail is consistent within this particular organisation. Due to the fact that it is computationally intensive and sometimes almost impossible to determine the scope on an email, we created a standardised form for each employee to complete. Each person having to postulate the persons which whom he/she interacts from within the company. Interactions were separated into 3 categories: academical, social or administrative. Based on the data collected we were able to create a graph for each type of interaction. These are also informal networks

as they are not enforced or regulated by the organisation.

The effectiveness of information transfer in networks shall be analysed with methods that are specific to Complex Networks science. By comparing the networks among them we shall determine which sources are more reliable, in order to use them for further studies. We are also interested in determining which are the key people in this process. Then we shall establish if the key persons are also *awarded* with high positions in the organisational chart. With the help of the information gathered in this study we plan to establish a strategy for enhancing the structure and communication in the organisation of interest.

4.4.1 Creating the networks

The organisational chart is the most encountered model of formal networks within an organisation. The chart is a hierarchical structure that follows the layout of a pyramid. Every individual in the organisation, except one, is subordinated to someone else in the organisation. Members communicate with their subordinates and with their respective superior. Such a network is perceived as the *right* way for knowledge exchange to take place.

As mentioned above the data needed for this paper was obtained from a small size academical institution, the Romanian-German University of Sibiu(Ro-Ger). The university has only two faculties: Economics and Law; The Economics department has more than double the personnel of the Law department. Overall, the institution has about 50 members, 31 individuals having academic attributions. Besides the academical staff, we shall include in the model of our networks 7 more people. This latter individuals play key roles in the functioning of the institution, being part of departments like: marketing, finances, secretarial, juridical and administrative; The difference up to 50 individuals is made by personnel with diverse attributions, e.g. cleaning, that are nonessential for our study. Therefore, each network shall contain 38 nodes. All the graphical interpretations of the networks are made by using *Force Atlas 2* graph layout in *Gephi* [19].

First of all, we shall be talking about the network based on the organisational chart (OrgNet). We started by "translating" each job position with their corresponding individuals. Then we linked each individual with his/her direct superior

and subordinate. Thus a link between two nodes represents a relation of subordination between them. The link is bidirectional as we considered that the exchange of information in this case comes from both sides. Fig. 4.4.1 is a graphical representation of the resulting network. Being a small institution some individuals are assigned with two or more roles, therefore they have multiple superiors.

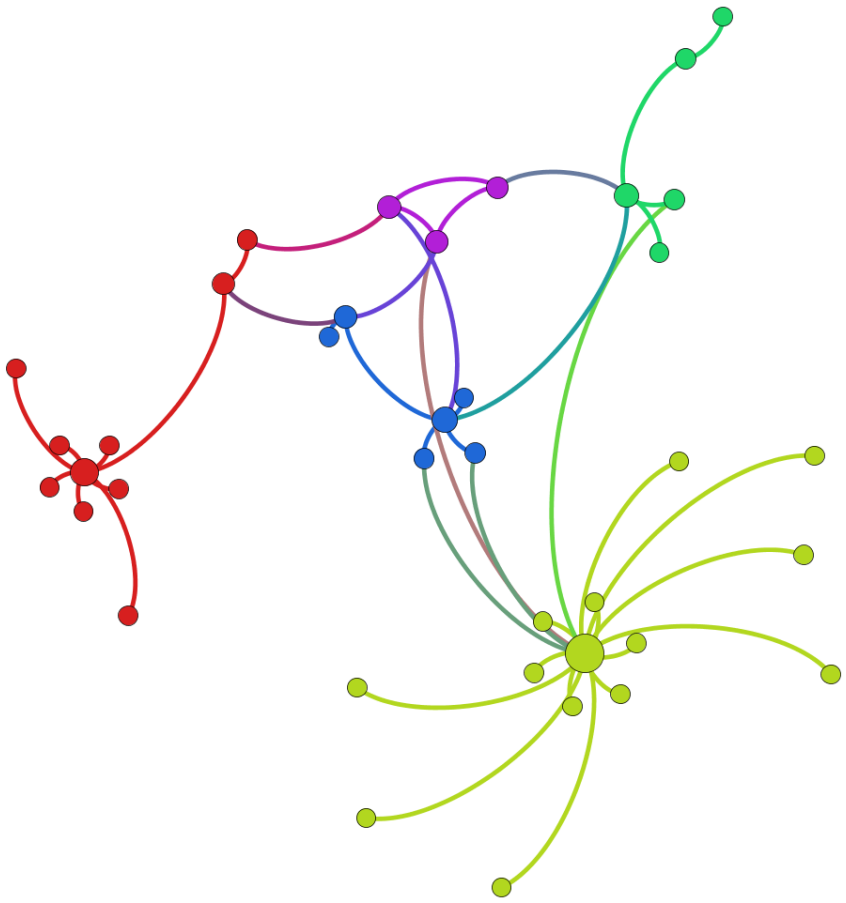


Figure 4.4.1: The network defined by the organisational chart. The size of each node is represented according to its Degree. Nodes' colour represent their corresponding communities, determined with the modularity algorithm. Red nodes represent some of the individuals in the Law faculty. Dark green nodes represent some of the people in the Economics faculty. The other communities are a mix of support personnel (marketing, finances, secretarial, juridical and administrative) and leadership individuals (rector, deans).

As we were granted access to email logs, we decided to use two time intervals that were of distinct length of time and from different academical periods. This will help us to figure out the information exchange in more than one scenario. We chose the first period of time to span from late February to late March, 33 days, as this represents the start of the second semester. The other period represents the end of the academical year from the middle of May to late June, 43 days. Also this decision was influenced by the fact that in the month of April due to the spring break email activity decreased. In order to create these graphs more than 4 million lines of log had to be parsed, as we had to eliminate: spam, email communication with the outside and other non essential data. Thus we obtained for the first period 376 emails and for the second period 525 emails.

For each of these periods we created a network, thus having two informal networks based on email logs. A link between two nodes was created if an email was sent between the corresponding individuals. The links are weighted, the weight represents the number of emails that correspond to that link. See Fig. 4.4.2 for the representation of the network in the first period of study. For the second period of time see Fig. 4.4.3. No communication has been found for several individuals in both periods of time, therefore their corresponding nodes have no links.

The third source of network creation was a form given to all the 38 individuals. Each individual had to specify the persons inside the university with whom he/she communicates. In Ro-Ger we could determine that every exchange of information could be categorised based on the subject into 3 types: academical, social and administrative; A number of 3 networks were constructed, one for each category of communication. Further more we aggregated these 3 networks in one network that represents all the 3 types of communication considered by us, called *Full Communication Network* (FCN). Links of this FCN have weights with values between 1 and 3. For example if person A and person B communicate in all 3 scenarios: social, administrative and academic; then the weight of the link between them will have a value of 3. Because of privacy issues only 20 out of 38 people completed the form. But the networks created from this source have all nodes connected, there is a path between every two nodes, as opposed to the networks created from the

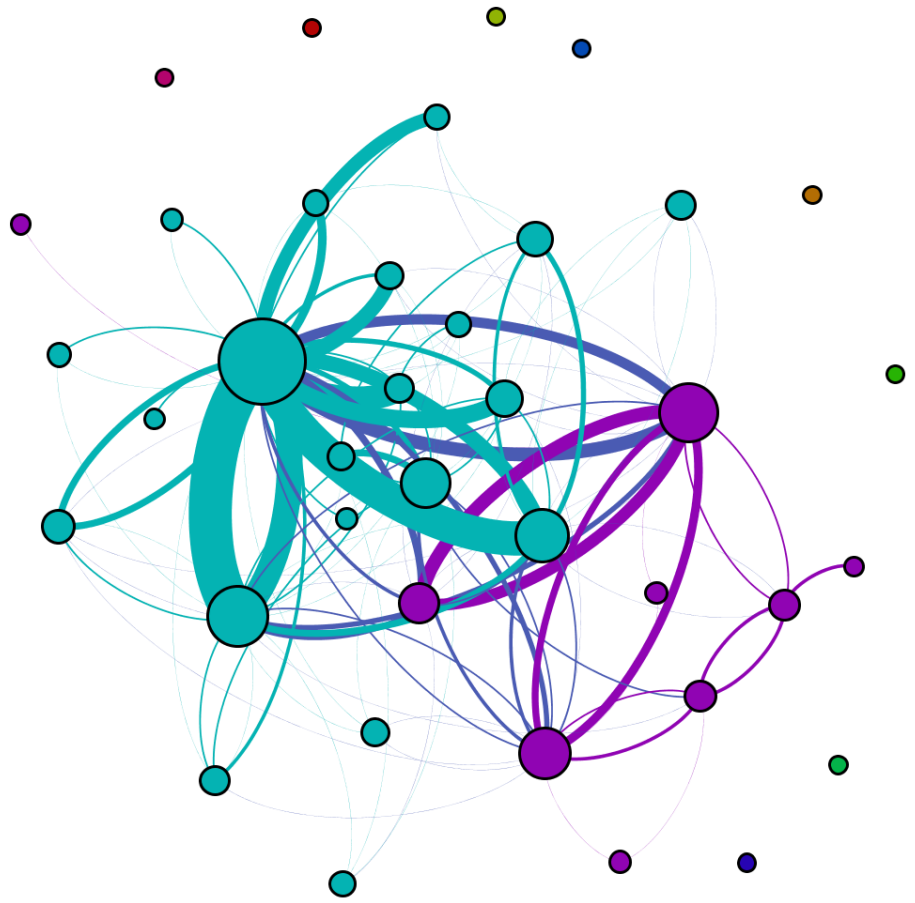


Figure 4.4.2: The network defined by the exchange of emails among the members of the academical organisation in the period February-March. The size of each node is represented according to its Degree. Nodes' colour represent their corresponding communities, determined with the modularity algorithm. The thickness of the links is proportional to their weight.

email logs. See Fig. 4.4.4 for the FCN and Fig. 4.4.5 for the network of social interactions.

In total we created 7 networks which will be analysed and compared among them in order to fulfil our goals: determine if the exchange of information inside the university is efficient and determine which sources of data are more reliable.

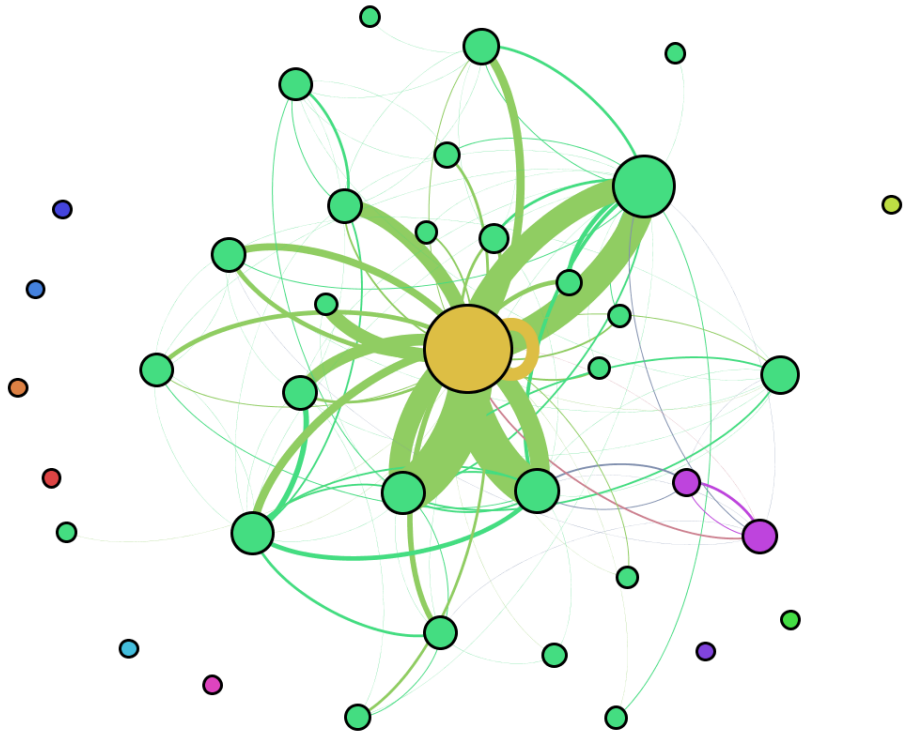


Figure 4.4.3: The network defined by the exchange of emails among the members of the academical organisation in the period May-June. The size of each node is represented according to its Degree. Nodes' colour represent their corresponding communities, determined with the modularity algorithm. The thickness of the links is proportional to their weight.

4.4.2 Discussions & Results

We shall present the results in a layered approach, as defined by Network Science. The metrics corresponding to each network shall be presented in parallel, thus making the comparisons easy to spot and understand.

The networks constructed could be interpreted from 2 different approaches: a directed graph or an undirected graph. The undirected graph approach captures communication that is not present in the email logs. We can argue that even if one

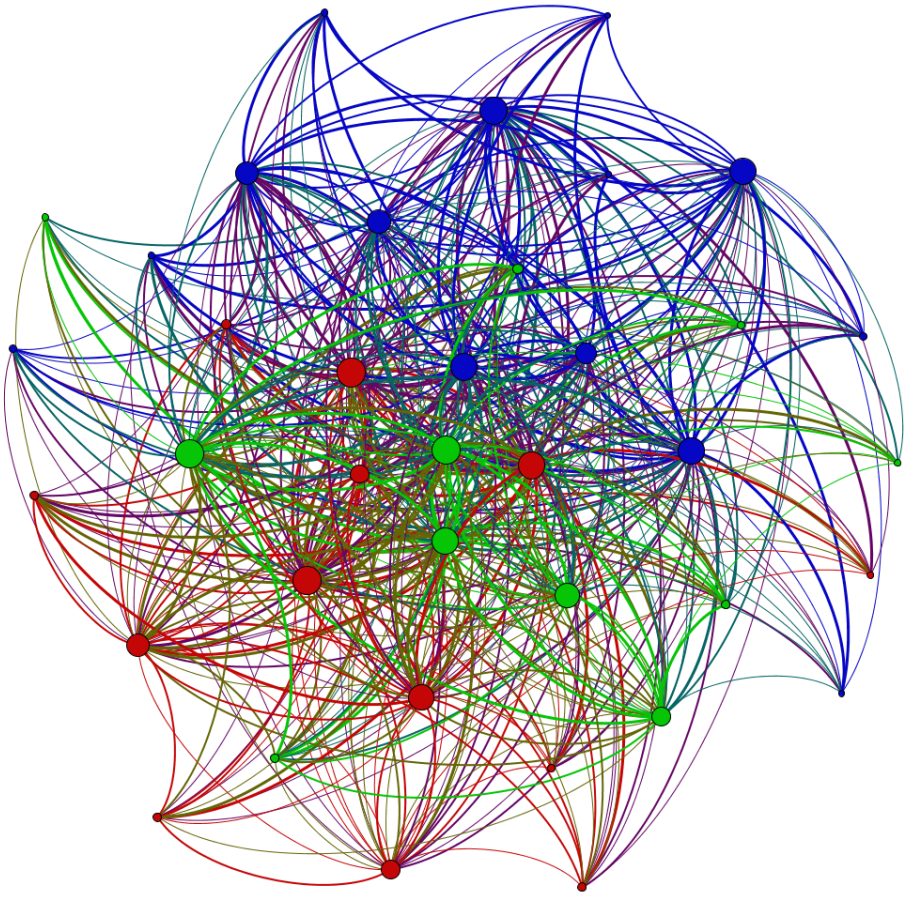


Figure 4.4.4: The network defined as the aggregated networks of: social, administrative and academical; interactions. The size of each node is represented according to its *Degree*. Nodes' colour represent their corresponding communities, determined with the modularity algorithm.

email was sent from A to B, then there should exist some communication channel (direct or indirect) by which B requested the email. For example we have person A which receives an email from B with his pay-check for the current month, both individuals work in the same organisation. Person A did not request that email directly or explicitly, he/she just gave his/her email address to the human resources department (HRD). The finance department, where B works, receives from the HRD all personnel contact addresses. Thus by considering the undirected graph approach we get a more complete picture of exchange of information within an

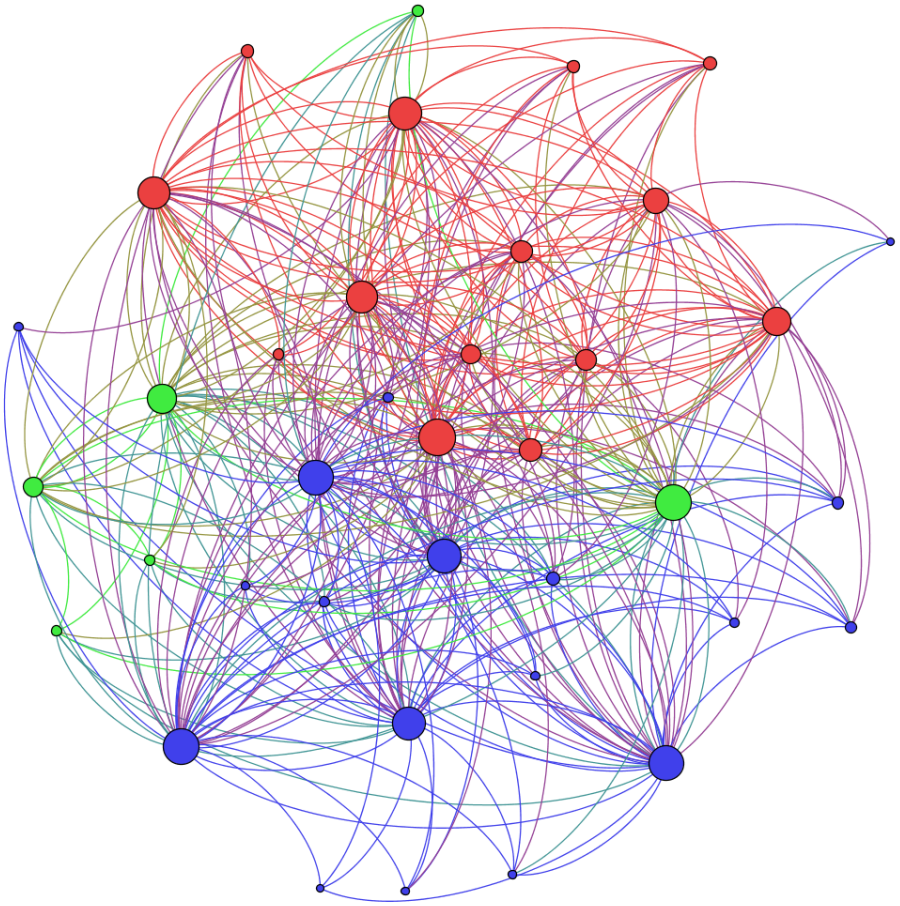


Figure 4.4.5: The network defined by the social interactions. The size of each node is represented according to its *Degree*. Nodes' color represent their corresponding communities, determined with the modularity algorithm.

organisation. Edge weights can be used to refine the metrics, making this network truly multidimensional.

For better understanding we shall call the network formed based on the organisational chart as *OrgNet*. The networks constructed based on email logs will be mentioned as *February-March* (FM) or *May-June* (MJ). In the case when we shall refer to both FM and MJ we shall name them as *MailNet*. The networks built from the data collected in forms shall me mentioned as: *Full Communication Network*

Network Metric	OrgNet	February March	May June	FCN	AC	Social	AD
Avg. Degree	2.316	3.316	2.738	15.447	8.842	10.5	10.263
Diameter	7	4	4	2	2	2	2
Modularity	0.562	0.154	0.024	0.058	0.129	0.098	0.092
Avg. Clustering Coefficient	0.048	0.4	0.627	0.848	0.761	0.833	0.825
Avg. Path Legth	3.492	2.007	1.933	1.367	1.6	1.568	1.543

Table 4.4.1: General graph description metrics. The *AC* column stands for the Academical network and the *AD* column stands for the Administrative network

Network Rank	OrgNet	February March	May June	FCN	Academical	Social	Administrative
1	h 28	a 26	a 26	h 28	d 15	a 26	d 15
2	h 7	d 11	d 11	d 27	d 25	a 17	a 26
3	h 4	a,d 19	a,d 19	a 26	a,d 19	h 28	d 27
4	h 12	a 17	a 17	d 15	d 27	d 11	a,d19
5	h 2	d 8	a 33	d 8	h 7	d 8	a 29

Table 4.4.2: Top 5 university employees by *PageRank* metric for each network constructed. Due to privacy issues the personnel will be mentioned by id. The letter that follows each id has the following meaning: *h* for a person with high position e.g. dean, *d* for persons with didactic responsibilities and no other rank, *a* for employees which have auxiliary responsibilities e.g. secretary;

(FCN), *Social, Academical* (AC), *Administrative* (AD); When talking in general about the last 4 networks we shall use the notation *FormNet*.

We start discussing the results by comparing metrics that characterise the entire graphs. You can see the values corresponding to each metric in Table 4.4.1. The *Average Degree* metric postulates the number of interactions with fellow colleagues of an university member. You can clearly see that there are important differences between all the sources of data. This being said we consider the *OrgNet* as the source with the most unrealistically feedback in this case. The differences between *MailNet* and *FormNet* can be explained by the shorter period of study in the case of *MailNet,* as *FormNet* is constructed based on all the interactions that could be remembered. When talking about the *Diameter OrgNet* has a very high value compared to the number of nodes. If this were the case then the exchange of information inside the university would be very fuzzy. But again the other 2

sources come to contradict *OrgNet*. The above conclusions for the *OrgNet* are sustained even when talking about the *Avg. Clustering Coefficient*, the value is unrealistically low. For example if person A has two friends B and C, then there is a high probability for B and C to be also friends, especially if A, B and C are in the same community. The *Average Clustering Coefficient* is used to determine the percentage of *Triadic Closure*. According to *OrgNet* there is almost no *Triadic Closure* (just one triangle), so only 3 university employees change information each with the other. For the *Average path length* we have to keep in mind the findings of V. Krebs [128] that if the value is larger than 3 there is a low probability for information to be exchanged.

The second layer of ONA analysis addresses community detection, which in our case is made with the *Modularity* method. In the case of *OrgNet* the *Modularity* metric value has the following interpretation: the university's employee are part of various littler clusters that rarely communicate among them; To be precise in *OrgNet* there are 6 communities with 3 to 12 members, see nodes' colour in Fig. 4.4.1. This is clearly not the case, see Fig. 4.4.4, where it is clear that there are 3 communities with some individuals exchanging more information than others. Two of these communities are formed by people in the Economics faculty and a third formed by members of the Law faculty. Also in the *MailNet* there are one or two large communities, with other communities formed by individual nodes that are not connected with the rest, see Fig. 4.4.2. At the end of the semester in MJ there is one big community (the nodes with the colour green), see Fig. 4.4.3, and a community made out of one person (the node with the yellow colour) that supports the communication of the big community. As it turns out the yellow node has the role of secretary, and the phenomena can be explained by the need to centralise students' grades and status before and during exams.

The third layer of ONA focuses on the metrics that define each node. The distribution of nodes' *Degree* in *OrgNet* and in the *MailNet* seems to follow a *power law* [1]. With the difference that the slope is milder in the *MailNet*. Pareto's distribution or *power law* distribution can be found in many other physical, biological and man-made phenomena. Also by looking at the *Weighted Degree* distribution in the *MailNet* we can find a *power law*. The presence of a *power law* distribution means that few members are responsible for a large part of the exchange of information.

For example in the *OrgNet* the top 7 individuals by *Degree*, representing 18% of the institutions' personnel, are responsible for 54% of connections. In *FormNet* the distribution of *Degree* is neither *power law* neither *normal*, see Fig. 4.4.6

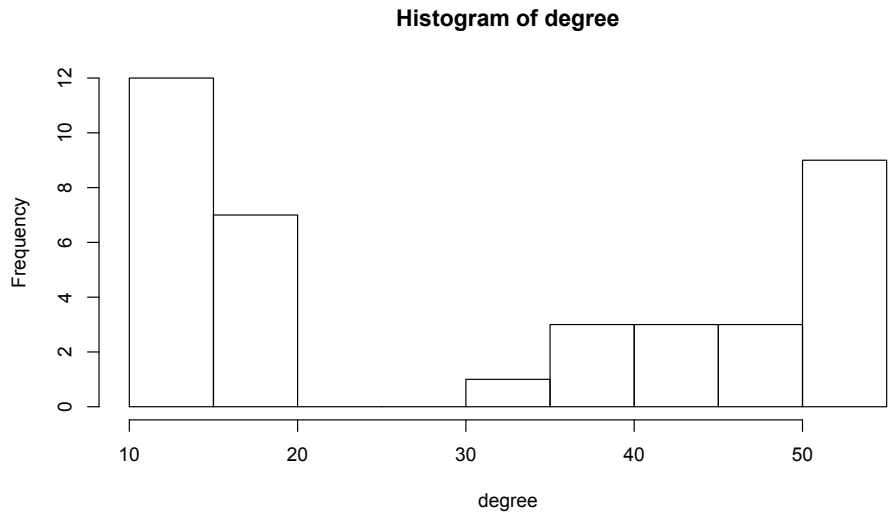


Figure 4.4.6: Distribution of *Degree* in FCN. On the X axis you have the value of the *Degree* and on the Y axis the number of nodes that have a certain *Degree*.

Another goal of this study was to determine alternative leaders in the university, employees that do not have any rank but play key roles in the exchange of information. By looking at Table 4.4.2 you can see that for *MailNet* and *FormNet* we discovered a large number of people that support the organisation but do not have high positions in the hierarchy. Due to the fact that *OrgNet* was created based on the organisational' chart the top 5 people by means of *PageRank* also have high positions in the university. Thus *OrgNet* can not be used to discover *alternative leaders*. If we take a further look at FM and MJ networks we can see that the people who communicate the most by email are in majority those that have auxiliary positions: secretary, IT personnel, etc.; If we compare the top persons by *PageRank* in FM and MJ we can see that few changes appear considering the fact that the periods of time are different. If we consider that in *MaiNet* there is a *power law* distribution of degree, we conclude that email is used all the time and extensively by

a small group of people. In AD, AC and *Social* there is no person that is common in the top 5 persons ranked by *PageRank*. In AD and AC person 19 and 27 are in common in the top 5, see Table 4.4.2. This results shows us that there is room for improving social skills with the persons that are ranked in top 5 of AC and AD.

4.5 Conclusions & Future Work

We can argue that in the case of knowledge dissipation and skill exchange SCRUM performs better than the classical hierarchical model. There is not a single Organisational Network Analysis metric where SCRUM will not outpace the tree-like structure. We conclude that ONA brings more evidence in favour of SCRUM or at least strengthens some of its already known advantages. Further we can say that ONA's techniques and metrics put in a easy to understand manner the differences between the two organisational models. We acknowledge that there is need for additional work in this field. Other agile development methods must be taken in consideration and interpreted in a network model. Also many other points of view must be used when constructing the network models besides the two used here. We take in consideration also the fact that the structure of a company is dependent on the organisation's field of work. Here we took the example of a software company but we need to broaden our space and take examples from other domains. Looking further down the road that this paper opened, there is a need for a more mathematical approach of this models.

Although our ontology presented in section 4.3 is in a primitive state we already prove it to be useful in a real life scenarios. Also the ontology is ready to be used as it is, and can be used in a wide variety of cases. Wherever a complex network is built, the ontology should be able to point to the correct insights on that network. Use across different fields of research was envisioned from the beginning thus we built an easy to use and understand ontology. Also we have supplied all the necessary means by which the ontology could be reached and used by putting it in a shared version controlled repository on *GitHub* [7]. We acknowledge that further work is needed with this ontology. For the future we plan to enrich it with domain specific classes that will guide professionals in better understanding real

[7] https://becheru.github.io/Ontology/

life events. Also we want to incorporate the necessary steps needed to be fulfilled in order to exchange the type of a network. Thus the ontology shall be also a guide of practice not only a type detection of complex networks. More examples of uses of the ontology in the real world need to be provided, together with SPARQL[8] examples.

Having compared some of the metrics from Table 4.4.1 we came to the following conclusions. The exchange of information within the RO-GER university si very good, affirmation supported by the metric values of *OrgNet* and *FormNet*. The organisational chart can not be considered a reliable source to approximate the information exchange within the university. We argue that *FormNet* returns the most reliable data. But two very important factors affect the precision of *FormNet*: it is time consuming to construct it and people may not wish to divulge the data needed. The *MailNet* does not suffer from the above mentioned problems but categorising the type of interactions may be difficult. Another problem of *MailNet* is that people may use several accounts for different purposes and that socialising through emails has become less and less used due to the widespread of social networks. Overall we could say that the *MailNet* is the best trade-off between time and precision.

Due to the analysis done by us we could get a clear view of how information is exchanged in the university. We plan to use this view in order to suggest a step by step plan for enhancing the exchange of information. First of all the plan should offer some means off supporting those employees that play key roles in the process of information exchange. Before our analysis many of them were unknown to give such great backing to the institution. Second of all we want to target the reunification of the personnel in the Economics faculty into one cluster based on the work done by *V. Krebs* [129]. For future simmilar studies we recommend that networks constructed for ONA should have different sources. As we observe in this paper ONA's conclusions are dependent on the source. We acknowledge that the data set for this study is small/very small, but due to the reticence of larger academic institutions we were not able to obtain larger data sets. With this study and further studies we are trying to *crack the ice* and understand better the academic environments, with the final purpose of better supporting knowledge exchange.

[8] Query language for RDF files [9]

To raise new questions, new possibilities, to regard old problems from a new angle, requires creative imagination and marks real advance in science.

Albert Einstein

5 Concluding Remarks

Throughout this dissertation we proposed a series of studies using Complex Networks Analysis techniques and methods on various fields of study. We argue that we have produced significant contributions in each of these fields as follows.

In Chapter 1 we have introduced a multi agent framework for the brokering of freight transport. We have developed and tested several ontologies in order to define the semantic modelling of information relevant to freight transportation and further improved on the means of communication and interaction among agents. Several real-life usage scenarios were conceived in order to validate and improve the ontologies in case. As freight brokers in our system are in a unique position of both buyers and sellers of transportation, which in many ways is in direct conflict with one another, we have studied the use negotiation mechanisms in order to resolve this conflict. We have proven through the use of a state or the negation protocol, i.e. ICNET, that negotiation mechanisms are capable of solving the conflict among the freight broker's interests. It is noteworthy to mention that ICNET was adapted for our scenario and agent personalities were introduced to better simu-

late real-world negotiations. Further more, a novel negotiation protocol, the Augmented Iterated Contract Net (AICNET), was developed introduced and tested through an in-house developed experimentation platform (publicly available on GitHub). The novelty of the protocol comes from the use Complex Networks Analysis inspired methods as means of providing the broker pertinent information that can used to employ complex dynamic business strategies, such as hindering the rise of monopolies. Through testing, developed and conducted by us, we have given compelling evidence that AICNET is superior to ICNET in hindering monopolies, mainly due to its broader understanding of the negotiation environments. Also, we can argue that AICNET has a truly dynamic nature since business strategies can be developed and changed with ease, as compared to ICNET.

In Chapter 2 we devoted our attention to achieving a better understating of collaborations that arise on a Romanian touristic internet portal, i.e. `http://amfostacolo.ro/`, through the use methods/techniques from various research fields such as: Complex Network Analysis, Natural Language Processing, Graph Databases and others; We have proven that the use Graph Databases such as *Neo4j* is relevant as it provides a capable querying and storage method, especially for touristic recommender systems. Through CNA methods we have discovered that the community on *AmFostAcolo* portal has meritocratic traits and that knowledge is exchanged is exchanged fast among other valuable insights. Thus we can consider the portal in case as a noteworthy information source. Moreover, we have introduced a novel method, based on CNA and NLP techniques, for text summarising via keyword identification and extraction, used for Romanian texts. The tests conducted have shown a superior performance of the proposed method in comparison with the widely used *tf-idf* information retrieval method. Also, we have provided evidence that CNA methods can be used to determine if textual reviews were automatically generated by software bots through inspecting the cohesion of networks generated from reviews.

In Chapter 3 we have presented two studies that relate to Social Learning Environments, having as case of study a course of Web Applications Development for 4th year undergraduate students which use several social platforms, i.e. Blogger, Twitter and MediaWiki, to collaborate in achieving their team oriented tasks. The first study uses textual complexity indices corroborated with the time evolu-

tion of students' posts on the social platforms in order to predict their academic performance. The statistical results have shown that the higher achieving students had several traits that could be used with success in predictive models, such as: the writing style, the number of unique words, the number of named entities and others. Similarly several textual indices were found to be good predictors for three of the four of Felder and Silverman's learning style models. The seconds study on social learning environments address the use of CNA methods to explore the collaborative learning environments that arise as part of the aforementioned course. A CNA based conceptual knowledge extraction framework for social learning environments was developed taken into account the pedagogical knowledge needs formulated by the course's instructors. We were able to provide compelling evidence on the validity of the framework by using it on the WAD course's data. Thus, the instructors were provided with valuable knowledge on students' collaboration at various granularity levels. Furthermore, we developed the *StudentViz* platform which provides the average computer literate instructors with some of framework's the knowledge extraction capabilities through graph visualisation.

In Chapter 4 we have tacked the use Complex Networks Analysis as means of evaluating organisational knowledge exchange. By using CNA metrics we were able to quantify for the first time the advantages of SCRUM's organisational model over the hierarchical model. Needles to say that the results are sustained only for the study case, a software product development scenario that requires 30 individuals, and for the mentioned assumptions. However, we can argue that in case of larger organisations the results for the hierarchical model will only be worse, due to the addition of several decisional layers. Next, we constructed an ontology based on complex networks traits that is able to infer the type of complex network. Knowing the type of a complex network provides those that are interested with many insights on the stability, knowledge exchange, social phenomena. The ontology was especially hard to develop since prior to it no other similar endeavours were attempted. To better grasp the value of this ontology, we can make a parallel with Medicine, the ontology brings to Complex Networks Analysis what a diseases categorisation method would bring to Medicine. Last, but not least an organisational analysis was conducted on a small sized academical organisation, the first analysis of its kind. The intended goal was to provide insights on a how individuals

involved in academia exchange knowledge. We found out that the organisational chart can not be considered as a reliable map on how information is exchanged, probably due to the liberal nature individuals involved in academia. Also, we discovered that the organisation did not support also individuals that were responsible for the good exchange of information. Generally speaking, the organisational study has proven once again the necessity of using CNA to discover and alleviate knowledge exchange in organisations.

It is noteworthy to consider that across this dissertation several methods from various Computer Science research branches were used in conjuncture with Complex Networks Analysis: Natural Language Processing, Ontologies, Databases and Web applications . Also, the research scenarios were various and diverse from freight transport to tourism to social learning environments to how organisations manage their work. Consider also the fact that Complex Networks Analysis is itself a multidisciplinary field comprising Computer Science, Physics, Mathematics and Social Sciences. Thus, we argue that the contributions presented in this dissertation have have a wide range of future possible outcomes.

References

[1] Lada A Adamic and Bernardo A Huberman. Power-law distribution of the world wide web. *science*, 287(5461):2115–2115, 2000.

[2] Jeffrey L Adler and Victor J Blue. A cooperative multi-agent transportation management and route guidance system. *Transportation Research Part C: Emerging Technologies*, 10(5):433–454, 2002.

[3] Malak Al-Hassan, Haiyan Lu, and Jie Lu. A semantic enhanced hybrid recommendation approach: A case study of e-government tourism service recommendation system. *Decision Support Systems*, 72:97–109, 2015.

[4] Kirsti Ala-Mutka et al. Review of learning in ict-enabled networks and communities. *Institute for Prospective Technological Studies*, 24061, 2009.

[5] Laura K Allen, Matthew E Jacovina, Mihai Dascalu, Rod D Roscoe, Kevin M Kent, Aaron D Likens, and Danielle S McNamara. {ENTER} ing the time series {SPACE}: Uncovering the writing process through keystroke analyses. *International Educational Data Mining Society*, 2016.

[6] Manish Arora and M Syamala Devi. Ontology based agent communication in resource allocation and monitoring. *International Journal of Computer Science Issues (IJCSI)*, 7(6):28, 2010.

[7] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25, 2000.

[8] National Governors Association et al. Common core state standards. *Washington, DC*, 2010.

[9] David Auber. Tulip—a huge graph visualization framework. In *Graph drawing software*, pages 105–126. Springer, 2004.

[10] Amelia Bădică and Costin Bădică. Formal verification of business processes as role activity diagrams. In *2011 Federated Conference on computer science and information systems (FedCSIS)*, pages 277–280. IEEE, 2011.

[11] Costin Bădică, Sorin Ilie, Michiel Kamermans, Gregor Pavlin, Ate Penders, Mihnea Scafeş, et al. Multi-agent systems, ontologies and negotiation for dynamic service composition in multi-organizational environmental management. *NATO Science for Peace and Security series D: Information and communication security*, (32):286–306, 2012.

[12] Costin Bădică, Mihaela Colhon, Alexandra Şendre, A Iftene, V Barbu Mititelu, D Cristea, and D Tufiş. Sentiment analysis of tourist reviews: Data preparation and preliminary results. In *Proceedings of the 10th International Conference Linguistic Resources And Tools For Processing The Romanian Language, ConsILR*, pages 135–142, 2014.

[13] Rodolfo Baggio. Network science and tourism–the state of the art. *Tourism Review*, 72(1):120–131, 2017.

[14] Rodolfo Baggio and Chris Cooper. Knowledge transfer in a tourism destination: the effects of a network structure. *The Service Industries Journal*, 30 (10):1757–1771, 2010.

[15] Ryan SJD Baker and Kalina Yacef. The state of educational data mining in 2009: A review and future visions. *JEDM| Journal of Educational Data Mining*, 1(1):3–17, 2009.

[16] Albert Bandura and Richard H Walters. *Social learning theory*, volume 1. Prentice-hall Englewood Cliffs, NJ, 1977.

[17] Albert-László Barabási. Scale-free networks: a decade and beyond. *science*, 325(5939):412–413, 2009.

[18] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[19] Mathieu Bastian, Sebastien Heymann, Mathieu Jacomy, et al. Gephi: an open source software for exploring and manipulating networks. *Icwsm*, 8 (2009):361–362, 2009.

[20] Vladimir Batagelj, Patrick Doreian, Natasa Kejzar, and Anuska Ferligoj. *Understanding large temporal networks and spatial networks: Exploration, pattern searching, visualization and network evolution*, volume 2. John Wiley & Sons, 2014.

[21] Alex Becheru. Agile development methods through the eyes of organisational network analysis. In *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14)*, page 53. ACM, 2014.

[22] Alex Becheru and Costin Bădică. Complex networks' analysis using an ontology-based approach: Initial steps. In *International Conference on Knowledge Science, Engineering and Management*, pages 326–337. Springer, 2014.

[23] Alex Becheru and Costin Bădică. A deeper perspective of online tourism reviews analysis using natural language processing and complex networks techniques. In *Proceedings of the 12th International Conference Linguistic Resources and Tools for Processing the Romanian Language, ConsILR*, pages 189–192, 2016.

[24] Alex Becheru and Elvira Popescu. Design of a conceptual knowledge extraction framework for a social learning environment based on social network analysis methods. In *2017 18th International Carpathian Control Conference (ICCC)*, pages 177–182. IEEE, 2017.

[25] Alex Becheru and Elvira Popescu. Using social network analysis to investigate students' collaboration patterns in emuse platform. In *2017 21st International Conference on System Theory, Control and Computing (ICSTCC)*, pages 266–271. IEEE, 2017.

[26] Alex Becheru, Costin Badica, and Mihaita Antonie. Complex network analysis of a tourism content sharing network. In *2015 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 407–414. IEEE, 2015.

[27] Alex Becheru, Costin Bădică, and Mihăiță Antonie. Towards social data analytics for smart tourism: a network science perspective. In *Workshop on Social Media and the Web of Linked Data*, pages 35–48. Springer, 2015.

[28] Alex Becheru, Florin Buşe, Mihaela Colhon, and Costin Bădică. Tourist review analytics using complex networks. In *Proceedings of the 7th Balkan Conference on Informatics Conference*, page 25. ACM, 2015.

[29] Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing multi-agent systems with JADE*, volume 7. John Wiley & Sons, 2007.

[30] Lawrence J Bendle and Ian Patterson. Network density, centrality, and communication in a serious leisure social world. *Annals of Leisure Research*, 11(1-2):1–19, 2008.

[31] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[32] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[33] Barry W Boehm. A spiral model of software development and enhancement. *Computer*, (5):61–72, 1988.

[34] Paolo Boldi and S Vigna. Axioms for centrality. *Internet Mathematics*, (Ga 288956):1–27, 2014. URL http://www.tandfonline.com/doi/abs/10.1080/15427951.2013.865686.

[35] Phillip Bonacich. Some unique properties of eigenvector centrality. *Social networks*, 29(4):555–564, 2007.

[36] Stephen P Borgatti and Martin G Everett. Models of core/periphery structures. *Social networks*, 21(4):375–395, 2000.

[37] Joan Borràs, Antonio Moreno, and Aida Valls. Intelligent tourism recommender systems: A survey. *Expert Systems with Applications*, 41(16):7370–7389, 2014.

[38] Ulrik Brandes. A faster algorithm for betweenness centrality. *Journal of mathematical sociology*, 25(2):163–177, 2001.

[39] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[40] Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. Graph structure in the web. *Computer networks*, 33(1-6):309–320, 2000.

[41] Amelia Bădică, Costin Bădică, Sorin Ilie, Alex Muscar, and Mihnea Scafeş. Dynamic selection of negotiation protocol in multi-agent systems for disaster management. In *Computational Collective Intelligence. Technologies and Applications - Third International Conference, ICCCI 2011, Gdynia, Poland, September 21-23, 2011, Proceedings, Part II*, pages 363–373, 2011.

[42] Costin Bădică and Mihnea Scafeş. One-to-many monotonic concession negotiation protocol. In *4th Balkan Conference in Informatics BCI2009*, pages 8–13. IEEE Computer Society Press, 2009.

[43] Costin Bădică and Mihnea Scafeş. Experiments with protocols for service negotiation. In *Proceeding of the Workshop on Applications of Software Agents – WASA 2011*, pages 25–31, 2011. ISBN 978-86-7031-188-6.

[44] Costin Bădică, Mihnea Scafeş, Sorin Ilie, Amelia Bădică, and Alex Muscar. Dynamic negotiations in multi-agent systems. *ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer*, page 8, 2011.

[45] Dimitrios Buhalis and Rob Law. Progress in information technology and tourism management: 20 years on and 10 years after the internet—the state of etourism research. *Tourism management*, 29(4):609–623, 2008.

[46] S Bull, A Duncan, B Ginon, and M Kickmeier-Rust. Educational data visualisation approaches and open learner modelling. *Project report*, 2015.

[47] Gülçin Büyüközkan and Buse Ergün. Intelligent system applications in electronic tourism. *Expert systems with applications*, 38(6):6586–6598, 2011.

[48] M Delgado Calvo-Flores, E Gibaja Galindo, MC Pegalajar Jiménez, and O Pérez Piñeiro. Predicting students' marks from moodle logs using neural network models. *Current Developments in Technology-Assisted Education*, 1(2):586–590, 2006.

[49] Mukun Cao, Xudong Luo, Xin Robert Luo, and Xiaopei Dai. Automated negotiation for e-commerce decision making: A goal deliberated agent architecture for multi-strategy selection. *Decision Support Systems*, 73:1–14, 2015.

[50] R Carter and F Bédard. E-business for tourism-practical guidelines for destination and businesses. *Madrid: World Tourism Organisation*, 2001.

[51] Bodong Chen, Yu-Hui Chang, Fan Ouyang, and Wanying Zhou. Fostering student engagement in online discussion through social learning analytics. *The Internet and Higher Education*, 37:21–30, 2018.

[52] Wai-Khuen Cheng and Huah-Yong Chan. *Conflict Resolution in Resource Federation with Intelligent Agent Negotiation*. INTECH Open Access Publisher, 2012.

[53] AK Chopra and MP Singh. Multiagent systems: A modern approach to distributed artificial intelligence. *Agent Communication*, pages 101–141, 2013.

[54] Gráinne Conole and Panagiota Alevizou. A literature review of the use of web 2.0 tools in higher education. *A report commissioned by the Higher Education Academy*, 2010.

[55] Pedro Terras Crespo and Cláudia Antunes. Predicting teamwork results from social network analysis. *Expert Systems*, 32(2):312–325, 2015.

[56] Rob Cross, Stephen P Borgatti, and Andrew Parker. Making invisible work visible: Using social network analysis to support strategic collaboration. *California management review*, 44(2):25–46, 2002.

[57] Rob Cross, Jane Linder, and Andrew Parker. Charged up: Managing the energy that drives innovation. *Management Quarterly*, 48(2):14, 2007.

[58] Robert L Cross, Jean Singer, Sally Colella, Robert J Thomas, and Yaarit Silverstone. *The organizational network fieldbook: Best practices, techniques and exercises to drive organizational innovation and performance*. John Wiley & Sons, 2010.

[59] Sakyibea Darko-Ampem and Maria Katsoufi. Towards a secure negotiation protocol for virtual organisations. *Master of Science Thesis, Department of Computer and Systems Sciences, Stockholm's University/Royal Institute of Technology*, 2006.

[60] Mihai Dascalu. Analyzing discourse and text complexity for learning and collaborating. *Studies in computational intelligence*, 534, 2014.

[61] Mihai Dascalu, Philippe Dessus, Maryse Bianco, Stefan Trausan-Matu, and Aurélie Nardy. Mining texts, learner productions and strategies with readerbench. In *Educational Data Mining*, pages 345–377. Springer, 2014.

[62] Mihai Dascalu, Stefan Trausan-Matu, Danielle S McNamara, and Philippe Dessus. Readerbench: Automated evaluation of collaboration based on cohesion and dialogism. *International Journal of Computer-Supported Collaborative Learning*, 10(4):395–423, 2015.

[63] Shane Dawson, Aneesha Bakharia, Elizabeth Heathcote, et al. Snapp: Realising the affordances of real-time sna within networked learning environments. Networked Learning, 2010.

[64] Pete Deemer, Gabrielle Benefield, Craig Larman, and Bas Vodde. The scrum primer. *Scrum Primer is an in-depth introduction to the theory and practice of Scrum, albeit primarily from a software development perspective, available at: http://assets. scrumtraininginstitute. com/downloads/1/scrumprimer121. pdf*, 1285931497:15, 2010.

[65] Peter DeGrace and Leslie Hulet Stahl. *Wicked Problems, Righteous Solutions: A Catalogue of Modern Engineering Paradigms*. Upper Saddle River, NJ: Prentice Hall, 1998.

[66] Michael Derntl and Sabine Graf. Impact of learning styles on student blogging behavior. In *2009 Ninth IEEE international conference on advanced learning technologies*, pages 369–373. IEEE, 2009.

[67] Ugur Dogrusoz, Erhan Giral, Ahmet Cetintas, Ali Civril, and Emek Demir. A layout algorithm for undirected compound graphs. *Information Sciences*, 179(7):980–994, 2009.

[68] Sergei N Dorogovtsev. *Lectures on complex networks*, volume 24. Oxford University Press Oxford, 2010.

[69] John Dron and Terry Anderson. *Teaching crowds: Learning and social media*. Athabasca University Press, 2014.

[70] Michal Drozdowicz, Katarzyna Wasielewska, Maria Ganzha, Marcin Paprzycki, Naoual Attaui, Ivan Lirkov, Richard Olejnik, Dana Petcu, and Costin Badica. Ontology for contract negotiations in agent-based grid resource management system. *Trends in parallel, distributed, grid and cloud computing for engineering*, pages 335–354, 2011.

[71] Michal Drozdowicz, Katarzyna Wasielewska, Maria Ganzha, Marcin Paprzycki, Naoual Attaui, Ivan Lirkov, Richard Olejnik, Dana Petcu, and Costin Badica. Ontology for contract negotiations in agent-based grid resource management system. *Trends in parallel, distributed, grid and cloud computing for engineering*, pages 335–354, 2011.

[72] Florence Dujardin, Kirstie Edwards, Sue Beckingham, C Cheal, J Coughlin, and S Moore. Diigo: social bookmarking in a professional communication course. *Transformation in Teaching: Social Media Strategies in Higher Education*, pages 243–274, 2012.

[73] David Easley and Jon Kleinberg. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.

[74] Ulle Endriss. Monotonic concession protocols for multilateral negotiation. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '06, pages 392–399. ACM, 2006.

[75] P ERDdS and A R&wi. On random graphs i. *Publ. Math. Debrecen*, 6:290–297, 1959.

[76] Gerald Falkowski and Valdis Krebs. Software test community uncovered using sna. *Orgnet. URL: http://www. orgnet. com/IBMCOPSNA. pdf [04-04-2011]*, 2005.

[77] Richard M Felder, Linda K Silverman, et al. Learning and teaching styles in engineering education. *Engineering education*, 78(7):674–681, 1988.

[78] Lijun Feng, Martin Jansche, Matt Huenerfauth, and Noémie Elhadad. A comparison of features for automatic readability assessment. In *Proceedings of the 23rd international conference on computational linguistics: Posters*, pages 276–284. Association for Computational Linguistics, 2010.

[79] Jacques Ferber and Gerhard Weiss. *Multi-agent systems: an introduction to distributed artificial intelligence*, volume 1. Addison-Wesley Reading, 1999.

[80] Raffaele Filieri and Fraser McLeay. E-wom and accommodation: An analysis of the factors that influence travelers' adoption of information from online reviews. *Journal of Travel Research*, 53(1):44–57, 2014.

[81] Tim Finin, Richard Fritzson, Don McKay, and Robin McEntire. Kqml as an agent communication language. In *Proceedings of the third international conference on Information and knowledge management*, pages 456–463. ACM, 1994.

[82] Foundation for Intelligent Physical Agents. Message structure specification, 2002. URL http://www.fipa.org/specs/fipa00061/SC00061G.html.

[83] Foundation for Intelligent Physical Agents. Fipa sl content language specification, 2002. URL http://www.fipa.org/specs/fipa00008/index.html.

[84] Foundation for Intelligent Physical Agents. Fipa communicative act library specification, 2004. URL http://www.fipa.org/specs/fipa00037/SC00037J.html.

[85] Nicoletta Fornara, Daniel Okouya, and Marco Colombetti. Using owl 2 dl for expressing acl content and semantics. In *European workshop on multi-agent systems*, pages 97–113. Springer, 2011.

[86] Mratin UML Fowler, Kendall Scott, et al. Distilled: A brief guide to the standard object modeling language, 2003.

[87] Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.

[88] Thomas MJ Fruchterman and Edward M Reingold. Graph drawing by force-directed placement. *Software: Practice and experience*, 21(11):1129–1164, 1991.

[89] Aldo Gangemi, Carola Catenacci, Massimiliano Ciaramita, and Jos Lehmann. A theoretical framework for ontology evaluation and validation. In *SWAP*, volume 166, page 16. Citeseer, 2005.

[90] Ángel García-Crespo, José Luis López-Cuadrado, Ricardo Colomo-Palacios, Israel González-Carrasco, and Belén Ruiz-Mezcua. Sem-fit: A semantic based expert system to provide recommendations in the tourism domain. *Expert systems with applications*, 38(10):13310–13319, 2011.

[91] Aitor García-Pablos, Montse Cuadros, and Maria Teresa Linaza. Automatic analysis of textual hotel reviews. *Information Technology & Tourism*, 16(1): 45–69, 2016.

[92] José M Gascueña, Elena Navarro, Patricia Fernández-Sotos, Antonio Fernández-Caballero, and Juan Pavón. Idk and icaro to develop multi-agent systems in support of ambient intelligence. *Journal of Intelligent and Fuzzy Systems*, 28:3–15, 2015.

[93] Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, and Grammati Pantziou. Mobile recommender systems in tourism. *Journal of network and computer applications*, 39:319–333, 2014.

[94] Maciej Gawinecki, Minor Gordon, Pawel Kaczmarek, and Marcin Paprzycki. The problem of agent-client communication on the internet. *Scalable Computing: Practice and Experience*, 6(1), 2005.

[95] Maciej Gawinecki, Maria Ganzha, Pawel Kobzdej, Marcin Paprzycki, Costin Badica, Mihnea Scafes, and Gabriel-George Popa. Managing information and time flow in an agent-based e-commerce system. In *2006 Fifth International Symposium on Parallel and Distributed Computing*, pages 352–359. IEEE, 2006.

[96] Vincenzo Gervasi, Vincenzo Ambriola, and ML Barbaresi. Quantitative assessment of textual complexity. *Complexity in language and text*, pages 197–228, 2002.

[97] Carlo Giovannella, Federico Scaccia, and Elvira Popescu. A pca study of student performance indicators in a web 2.0-based learning environment. In *2013 IEEE 13th International Conference on Advanced Learning Technologies*, pages 33–35. IEEE, 2013.

[98] Ulrike Gretzel, Hannes Werthner, Chulmo Koo, and Carlos Lamsfus. Conceptual foundations for understanding smart tourism ecosystems. *Computers in Human Behavior*, 50:558–563, 2015.

[99] Arent Greve, Mario Benassi, and Arne Dag Sti. Exploring the contributions of human and social capital to productivity. *International Review of Sociology*, 20(1):35–58, 2010.

[100] Arent Greve, Mario Benassi, and Arne Dag Sti. Exploring the contributions of human and social capital to productivity. *International Review of Sociology*, 20(1):35–58, 2010.

[101] Jerrold W Grossman and Patrick DF Ion. On a portion of the well-known collaboration graph. *Congressus Numerantium*, pages 129–132, 1995.

[102] Thomas R Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5-6):907–928, 1995.

[103] Nicola Guarino. *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*, volume 46. IOS press, 1998.

[104] Javier Gutierrez, Juan Carlos Garcia-Palomares, Gustavo Romanillos, and Maria Henar Salas-Olmedo. Airbnb in tourist cities: comparing spatial patterns of hotels and peer-to-peer accommodation. *arXiv preprint arXiv:1606.07138*, 2016.

[105] Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[106] Caroline Haythornthwaite. Social network methods and measures for examining e-learning. *Social networks*, 2005:1–22, 2005.

[107] Caroline Haythornthwaite and Maarten De Laat. Social networks and learning networks: Using social network perspectives to understand social learning. In *Proceedings of the 7th international conference on networked learning*, pages 183–190. Lancaster University Aalborg, Denmark, 2010.

[108] Martin Hepp. Ontologies: State of the art, business potential, and grand challenges. In *Ontology Management*, pages 3–22. Springer, 2008.

[109] Martin Hepp. Goodrelations: An ontology for describing products and services offers on the web. In *International Conference on Knowledge Engineering and Knowledge Management*, pages 329–346. Springer, 2008.

[110] Daniel A Hojman and Adam Szeidl. Core and periphery in networks. *Journal of Economic Theory*, 139(1):295–309, 2008.

[111] Matthew Horridge, Holger Knublauch, Alan Rector, Robert Stevens, and Chris Wroe. A practical guide to building owl ontologies using the protégé-owl plugin and co-ode tools edition 1.0. *University of Manchester*, 2004.

[112] Matthew Horridge, Nick Drummond, John Goodwin, Alan L Rector, Robert Stevens, and Hai Wang. The manchester owl syntax. In *OWLed*, volume 216, 2006.

[113] Florentina Hristea and Mihaela Colhon. Feeding syntactic versus semantic knowledge to a knowledge-lean unsupervised word sense disambiguation algorithm with an underlying naïve bayes model. *Fundamenta Informaticae*, 119(1):61–86, 2012.

[114] Florentina Hristea and Marius Popescu. A dependency grammar approach to syntactic analysis with special reference to romanian. *Building Awareness in Language Technology. University of Bucharest Publishing House*, 2003.

[115] Florentina T Hristea. *The Naïve Bayes Model for Unsupervised Word Sense Disambiguation: Aspects Concerning Feature Selection*. Springer Science & Business Media, 2012.

[116] Chun-Che Huang, Wen-Yau Liang, Yu-Hsin Lai, and Yin-Chen Lin. The agent-based negotiation process for b2c e-commerce. *Expert Systems with Applications*, 37(1):348–359, 2010.

[117] Sorin Ilie, Costin Bădică, Amelia Bădică, Liviu Sandu, Raluca Sbora, Maria Ganzha, and Marcin Paprzycki. Information flow in a distributed agent-based online auction system. In *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics*, page 42. ACM, 2012.

[118] Peter Jackson and Isabelle Moulinier. *Natural language processing for online applications*. John Benjamins, 2007.

[119] Mathieu Jacomy, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software. *PloS one*, 9(6):e98679, 2014.

[120] Nick R Jennings, Peyman Faratin, Alessio R Lomuscio, Simon Parsons, Michael J Wooldridge, and Carles Sierra. Automated negotiation: prospects, methods and challenges. *Group Decision and Negotiation*, 10(2): 199–215, 2001.

[121] Kai Jiang, Huagang Yin, Peng Wang, and Nenghai Yu. Learning from contextual information of geo-tagged web photos to rank personalized tourism attractions. *Neurocomputing*, 119:17–25, 2013.

[122] Chen Jinyan, Becken Susanne, and Stantic Bela. Citizen science–exploring tourists' observations on micro blogs as a tool to monitor environmental change.

[123] Reynol Junco, Greg Heiberger, and Eric Loken. The effect of twitter on college student engagement and grades. *Journal of computer assisted learning*, 27(2):119–132, 2011.

[124] Bill Karakostas and Dimitris K Kardaras. A knowledge graph for travel mode recommendation and critiquing. *DBKDA 2017*, page 87, 2017.

[125] Leo Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[126] James W Keefe. Learning style: An overview. *Student learning styles: Diagnosing and prescribing programs*, 1:1–17, 1979.

[127] Jon M Kleinberg. Hubs, authorities, and communities. *ACM computing surveys (CSUR)*, 31(4es):5, 1999.

[128] Valdis Krebs. Managing the 21st century organization. *ihrim Journal*, 11 (4):2–8, 2007.

[129] Valdis Krebs and June Holley. Building smart communities through network weaving. *Appalachian Center for Economic Networks*, 2006.

[130] Sandra Kübler, Ryan McDonald, and Joakim Nivre. Dependency parsing. *Synthesis Lectures on Human Language Technologies*, 1(1):1–127, 2009.

[131] Renaud Lambiotte, J-C Delvenne, and Mauricio Barahona. Laplacian dynamics and multiscale modular structure in networks. *arXiv preprint arXiv:0812.1770*, 2008.

[132] Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117, 2009.

[133] Thomas K Landauer and Susan T Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211, 1997.

[134] Matthieu Latapy. Main-memory triangle computations for very large (sparse (power-law)) graphs. *Theoretical computer science*, 407(1-3):458–473, 2008.

[135] Fernando Lopes and Helder Coelho, editors. *Negotiation and Argumentation in Multi-Agent Systems. Fundamentals, Theories, Systems and Applications*. Bentham Science Publishers, 2014. ISBN 978-1-60805-825-9. doi: 10.2174/97816080582421140101. URL http://ebooks.benthamscience.com/book/9781608058242/.

[136] Joanne S Luciano, Bosse Andersson, Colin Batchelor, Olivier Bodenreider, Tim Clark, Christine K Denney, Christopher Domarew, Thomas Gambet, Lee Harland, Anja Jentzsch, et al. The translational medicine ontology and knowledge base: driving personalized medicine by bridging the gap between bench and bedside. In *Journal of biomedical semantics*, volume 2, page S1. BioMed Central, 2011.

[137] Lucian Luncean and Costin Bădică. Semantic modeling of information for freight transportation broker. In *Proceeding of 16th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing – SYNASC 2014*, pages 527––534. IEEE Computer Society, 2014.

[138] Lucian Luncean, Costin Bădică, and Amelia Bădică. Agent-based system for brokering of logistics services – initial report. In Ngoc Thanh Nguyen, Boonwat Attachoo, Bogdan Trawinski, and Kulwadee Somboonviwat, editors, *Proc. Intelligent Information and Database Systems – ACIIDS 2014, Part II*, volume 8398 of *Lecture Notes in Computer Science*, pages 485–494. Springer, 2014.

[139] Lucian Luncean, Alex Becheru, and Costin Bădică. Initial evaluation of an ontology for transport brokering. In *Computer Supported Cooperative Work in Design (CSCWD), 2015 IEEE 19th International Conference on*, pages 121–126. IEEE, 2015.

[140] Lucian Luncean, Alex Becheru, and Costin Bădică. Initial evaluation of an ontology for transport brokering. In *Proceeding of 19th IEEE International Conference on Computer Supported Cooperative Work in Design – CSCWD 2015*, 2015.

[141] Lucian Luncean, Andrei Mocanu, and Alexandru Petrut Becheru. Automated negotiation framework for the transport logistics service. In *Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2016 18th International Symposium on*, pages 387–394. IEEE, 2016.

[142] David Lusseau. The emergent properties of a dolphin social network. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 270 (suppl_2):S186–S188, 2003.

[143] Seid Maglajlić et al. Efficiency in e-learning: Can learning outcomes be improved by using social networks of trainees and tutors? *Economics, Management, and Financial Markets*, 7(4):121–137, 2012.

[144] Andreia Malucelli and Eugénio da Costa Oliveira. Ontology-services to facilitate agents' interoperability. In *Pacific Rim International Workshop on Multi-Agents*, pages 170–181. Springer, 2003.

[145] Christopher D Manning, Christopher D Manning, and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT press, 1999.

[146] Edison Marrese-Taylor, Juan D Velásquez, Felipe Bravo-Marquez, and Yutaka Matsuo. Identifying customer preferences about tourism products using an aspect-based opinion mining approach. *Procedia Computer Science*, 22:182–191, 2013.

[147] John S McGee. Predatory price cutting: the standard oil (nj) case. *The Journal of Law and Economics*, 1:137–169, 1958.

[148] W McKnight. Chapter twelve–graph databases: when relationships are the data. *Information Management*, pages 120–131, 2014.

[149] Danielle S McNamara, AC Graesser, and MM Louwerse. Sources of text difficulty: Across genres and grades. *Measuring up: Advances in how we assess reading ability*, pages 89–116, 2012.

[150] Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1): 415–444, 2001.

[151] Martijn Mes, Matthieu van der Heijden, and Peter Schuur. Interaction between intelligent agent strategies for real-time transportation planning. *Central European journal of operations research*, 21(2):337–358, 2013.

[152] Paula Gomes Mian and Ricardo de Almeida Falbo. Supporting ontology development with oded. *Journal of the Brazilian Computer Society*, 9(2): 57–76, 2003.

[153] JIL Miguéns and JFF Mendes. Travel and tourism: Into a complex network. *Physica A: Statistical Mechanics and its Applications*, 387(12):2963–2971, 2008.

[154] Rada Mihalcea. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, 2004.

[155] Stanley Milgram. The small world problem. *Psychology today*, 2(1):60–67, 1967.

[156] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[157] J Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1): 415–444, 2001.

[158] Through Educational Data Mining. Enhancing teaching and learning through educational data mining and learning analytics: An issue brief. In *Proceedings of conference on advanced technology for education*, 2012.

[159] R Bruce Money. International multilateral negotiations and social networks. *Journal of International Business Studies*, 29(4):695–710, 1998.

[160] Susan Allen Nan. Conflict resolution in a network society. *International Negotiation*, 13(1):111–131, 2008.

[161] Roberto Navigli and Mirella Lapata. An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE transactions on pattern analysis and machine intelligence*, 32(4):678–692, 2010.

[162] Jessica Nelson, Charles Perfetti, David Liben, and Meredith Liben. Measures of text difficulty: Testing their predictive value for grade levels and student performance. *Council of Chief State School Officers, Washington, DC*, 2012.

[163] Mark EJ Newman. Clustering and preferential attachment in growing networks. *Physical review E*, 64(2):025102, 2001.

[164] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[165] Mark EJ Newman. Modularity and community structure in networks. *Proceedings of the national academy of sciences*, 103(23):8577–8582, 2006.

[166] Dan Novak, Mark Rennaker, and Paulette Turner. Using organizational network analysis to improve integration across organizational boundaries. *People and Strategy*, 34(4):32, 2011.

[167] Natalya F Noy, Deborah L McGuinness, et al. Ontology development 101: A guide to creating your first ontology, 2001.

[168] Natalya F Noy, Deborah L McGuinness, et al. Ontology development 101: A guide to creating your first ontology, 2001.

[169] L Obrst, C Werner, M Inderjeet, R Steve, and B Smith. The evaluation of ontologies: Toward improved semantic interoperability. dans jo christopher, baker, & k.-h. cheung. *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*, 2007.

[170] Tihomir Orehovački, Goran Bubaš, and Andreja Kovačić. Taxonomy of web 2.0 applications with educational potential. *Transformation in teaching: Social media strategies in higher education*, pages 43–72, 2012.

[171] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

[172] Ellis B Page. The use of the computer in analyzing student essays. *International review of education*, 14(2):210–225, 1968.

[173] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[174] Zachary A Pardos, Neil T Heffernan, Brigham Anderson, and Cristina L Heffernan. The effect of model granularity on student performance prediction using bayesian networks. In *International Conference on User Modeling*, pages 435–439. Springer, 2007.

[175] Kathryn Pavlovich. The evolution and transformation of a tourism destination network: the waitomo caves, new zealand. *Tourism Management*, 24 (2):203–216, 2003.

[176] Auliana Poon et al. *Tourism, technology and competitive strategies*. CAB international, 1993.

[177] Elvira Popescu. *Social learning environments*. Sitech, 2013.

[178] Elvira Popescu. Providing collaborative learning support with social media in an integrated environment. *World Wide Web*, 17(2):199–212, 2014.

[179] Elvira Popescu and Florin Leon. Analyzing the influence of learning styles on students' behavior in a social learning environment. *Annals of the University of Craiova, Automation, Computers, Electronics and Mechatronics Series*, 11(38):1, 2014.

[180] Stefan Poslad. Specifying protocols for multi-agent systems interaction. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 2(4):15, 2007.

[181] Stefan Poslad. Specifying protocols for multi-agent systems interaction. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 2(4):15, 2007.

[182] Richard A Posner. *Antitrust law*. University of Chicago press, 2009.

[183] Angelo Presenza and Maria Cipollina. Analysis of links and features of tourism destination's stakeholders. an empirical investigation of a south italian region. 2009.

[184] Reihaneh Rabbany, Samira ElAtia, Mansoureh Takaffoli, and Osmar R Zaïane. Collaborative learning of students in online discussion forums: A social network analysis perspective. In *Educational data mining*, pages 441–466. Springer, 2014.

[185] Dragomir R Radev and Rada Mihalcea. Networks and natural language processing. *AI magazine*, 29(3):16–16, 2008.

[186] Daniel Roberge, Anthony Rojas, and Ryan Baker. Does the length of time off-task matter? In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge*, pages 234–237. ACM, 2012.

[187] Ian Robinson, Jim Webber, and Emil Eifrem. *Graph databases*. " O'Reilly Media, Inc.", 2013.

[188] MC Rodriguez-Sanchez, Juan Martinez-Romo, Susana Borromeo, and JA Hernandez-Tamames. Gat: Platform for automatic context-aware mobile services for m-tourism. *Expert Systems with applications*, 40(10):4154–4163, 2013.

[189] Cristóbal Romero and Sebastián Ventura. Educational data mining: a review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(6):601–618, 2010.

[190] Cristóbal Romero, Sebastián Ventura, Pedro G Espejo, and César Hervás. Data mining algorithms to classify students. In *Educational data mining 2008*, 2008.

[191] Cristóbal Romero, Manuel-Ignacio López, Jose-María Luna, and Sebastián Ventura. Predicting students' final performance from participation in online discussion forums. *Computers & Education*, 68:458–472, 2013.

[192] Gillian M Sandstrom and Elizabeth W Dunn. Social interactions and well-being: The surprising power of weak ties. *Personality and Social Psychology Bulletin*, 40(7):910–922, 2014.

[193] Mihnea Scafeş and Costin Bădică. *Complex Negotiations in Multi-Agent Systems*, volume 2 of *Automation, Computers, Electronics and Mechatronics*. Universitaria Publishing House, 2010. URL http://ace.ucv.ro/anale/2010_vol2/09MihneaScafes.pdf.

[194] Thomas Schank and Dorothea Wagner. *Approximating clustering-coefficient and transitivity*. Universität Karlsruhe, Fakultät für Informatik, 2004.

[195] Andreas Scheuermann. Logistics ontologies primer and user's guide. *InterLogGrid-Konsortium, Tech. Rep*, 2011.

[196] John Scott. *Social network analysis*. Sage, 2017.

[197] John R Searle. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press, 1969.

[198] Scott E Seibert, Maria L Kraimer, and Robert C Liden. A social capital theory of career success. *Academy of management journal*, 44(2):219–237, 2001.

[199] Claude E Shannon. Prediction and entropy of printed english. *Bell system technical journal*, 30(1):50–64, 1951.

[200] Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504, 2003.

[201] Marianna Sigala and Kalotina Chalkiti. Investigating the exploitation of web 2.0 for knowledge management in the greek tourism industry: An utilisation–importance analysis. *Computers in Human Behavior*, 30:800–812, 2014.

[202] Radu Simionescu. Hybrid pos tagger. In *Proceedings of Language Resources and Tools with Industrial Applications Workshop (Eurolan 2011 Summer School), Cluj-Napoca, Romania*, pages 21–28. Citeseer, 2011.

[203] Henry B Slotnick. Toward a theory of computer essay grading. *Journal of Educational Measurement*, 9(4):253–263, 1972.

[204] R.G. Smith. The contract net protocol: Highlevel communication and control in a distributed problem solver. In *IEEE Trans. Comput., 29(12)*, pages 1104–1113, 1980.

[205] Barbara A Soloman and Richard M Felder. Index of learning styles questionnaire. *NC State University. Available online at: http://www. engr. ncsu. edu/learningstyles/ilsweb. html (last visited on 14.05. 2010)*, 70, 2005.

[206] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

[207] Steven H Strogatz. Exploring complex networks. *nature*, 410(6825):268, 2001.

[208] Jeff Sutherland and Ken Schwaber. The scrum guide. *The definitive guide to scrum: The rules of the game. Scrum. org*, 268, 2013.

[209] Karolina Taczanowska, Luis-Millán González, Xavier Garcia-Massó, Andreas Muhar, Christiane Brandenburg, and José-Luis Toca-Herrera. Evaluating the structure and use of hiking trails in recreational areas using a mixed gps tracking and graph theory approach. *Applied Geography*, 55: 184–192, 2014.

[210] Hirotaka Takeuchi and Ikujiro Nonaka. The new new product development game. *Harvard business review*, 64(1):137–146, 1986.

[211] Sarosh N Talukdar. Collaboration rules for autonomous software agents. *Decision Support Systems*, 24(3-4):269–278, 1999.

[212] Valentina Tamma, Stephen Cranefield, Timothy W Finin, and Steven Willmott. *Ontologies for agents: Theory and experiences*. Springer Science & Business Media, 2005.

[213] Eunice Tang and Cherlotte Lam. Building an effective online learning community (olc) in blog-based teaching portfolios. *The Internet and Higher Education*, 20:79–85, 2014.

[214] Stefan Trausan-Matu, Mihai Dascalu, and Philippe Dessus. Textual complexity and discourse structure in computer-supported collaborative learning. In *International Conference on Intelligent Tutoring Systems*, pages 352–357. Springer, 2012.

[215] Jeffrey Travers and Stanley Milgram. An experimental study of the small world problem. In *Social Networks*, pages 179–197. Elsevier, 1977.

[216] Wei-Tek Tsai, Wu Li, Jay Elston, and Yinong Chen. Collaborative learning using wiki web sites for computer science undergraduate education: A case study. *IEEE Transactions on Education*, 54(1):114–124, 2011.

[217] Blanka Tundys, Andrzej Rzeczycki, Magdalena Zioło, and Jarosław Jankowski. Models of an integrated performance measurement system of intelligent and sustainable supply chains. In *Asian Conference on Intelligent Information and Database Systems*, pages 504–514. Springer, 2014.

[218] Mike Uschold and Michael Gruninger. Ontologies: Principles, methods and applications. *The knowledge engineering review*, 11(2):93–136, 1996.

[219] Chris Van Aart, Ruurd Pels, Giovanni Caire, and Federico Bergenti. Creating and using ontologies in agent communication. In *Proceedings of the Workshop on Ontologies in Agent Systems*, 2002.

[220] Egbert van der Zee and Dominique Vanneste. Tourism networks unravelled; a review of the literature on networks in tourism management studies. *Tourism Management Perspectives*, 15:46–56, 2015.

[221] Teun Adrianus Van Dijk, Walter Kintsch, and Teun Adrianus Van Dijk. Strategies of discourse comprehension. 1983.

[222] Quentin Van Doosselaere. *Commercial agreements and social dynamics in medieval Genoa*. Cambridge University Press, 2009.

[223] Katrien Verbert, Erik Duval, Joris Klerkx, Sten Govaerts, and José Luis Santos. Learning analytics dashboard applications. *American Behavioral Scientist*, 57(10):1500–1509, 2013.

[224] Denny Vrandečić. Ontology evaluation. In *Handbook on ontologies*, pages 293–313. Springer, 2009.

[225] Junli Wang, Zhijun Ding, and Changjun Jiang. An ontology-based public transport query system. In *2005 First International Conference on Semantics, Knowledge and Grid*, pages 62–62. IEEE, 2005.

[226] Katarzyna Wasielewska, Maria Ganzha, Marcin Paprzycki, Michal Drozdowicz, Dana Petcu, Costin Badica, Naoual Attaoui, Ivan Lirkov, and Richard Olejnik. Negotiations in an agent-based grid resource brokering system, 2011.

[227] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440, 1998.

[228] Danny Weyns, Nelis Boucké, Tom Holvoet, and Bart Demarsin. Dyncnet: A protocol for dynamic task assignment in multiagent systems, 2008.

[229] Dominic Widdows and Beate Dorow. A graph model for unsupervised lexical acquisition. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.

[230] William Wresch. The imminence of grading essays by computer–25 years later. *Computers and composition*, 10(2):45–58, 1993.

[231] Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics, 1994.

[232] Wanli Xing, Rui Guo, Eva Petakovic, and Sean Goggins. Participation-based student final performance prediction model through interpretable genetic programming: Integrating learning analytics, educational data mining and theory. *Computers in Human Behavior*, 47:168–181, 2015.

[233] Ji Soo Yi, Youn-ah Kang, John T Stasko, and Julie A Jacko. Understanding and characterizing insights: how do people gain insights using information visualization? In *Proceedings of the 2008 Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization*, page 4. ACM, 2008.

[234] Jaebong Yoo and Jihie Kim. Can online discussion participation predict group project performance? investigating the roles of linguistic features and participation patterns. *International Journal of Artificial Intelligence in Education*, 24(1):8–32, 2014.

[235] Hua Yuan, Hualin Xu, Yu Qian, and Yan Li. Make your travel smarter: Summarizing urban tourism information from massive blog data. *International Journal of Information Management*, 36(6):1306–1319, 2016.

[236] Amelia Zafra and Sebastián Ventura. Predicting student grades in learning management systems with multiple instance genetic programming. *International Working Group on Educational Data Mining*, 2009.

A

Curiculum Vitae

europass

Personal information

Surname(s) / First name(s)	**Alexandru-Petruţ / Becheru**
Address(es)	Str. Aurel Vlaicu, Nr. 12, Bl. 144, Ap. 5, Craiova, RO-200575, Romania
Email(s)	becheru@gmail.com
Nationality(-ies)	Romanian
Date of birth	April 1989
Gender	male
Web-Site	becheru.net

Work experience

Date	10.2014 – 09.2018
Occupation or position held	Teaching Assistant, University of Craiova, Romania
Laboratories held	Graph algorithms, Information Retrieval Algorithms, Programming techniques, C programming, Formal Languages & Automata
Date	10.2018 – Currently
Occupation or position held	Software Engineer, CS Communication & Systems Romania
Job Synopsis	Designer and integrator of mission critical systems

Education and training

Institution	"Stefan Odobleja" Computer Science High School, Craiova, Romania
Period	2004-2008
Institution	Faculty of Mathematics and Computer Science, University of Craiova, Romania
Period	2008-2011
Title of qualification awarded	B.Sc. in Computer Science
Institution	Faculty of Mathematics and Computer Science, University of Craiova, Romania
Period	2011 – 2013
Title of qualification awarded	M.Sc. in Artificial Intelligence
Institution	Universita degli Studi Roma 3, Rome, Italy
Period	October 2012 – March 2013
Title of qualification awarded	Erasmus academic exchange

Institution	Faculty of Automation Computers & Electronics ,University of Craiova, Romania
Period	2013 – Currently
Title of qualification awarded	Ph.D candidate in Network Science/Complex Networks

Personal skills and competences

Mother tongue(s) **Romanian**

Other language(s) English, French, Italian

Self-assessment *European level*(*)	**Understanding**		**Speaking**		**Writing**
	Listening	Reading	Spoken interaction	Spoken production	
English	C1	C2	C1	C2	C2
French	B1	B2	B1	B2	B2
Italian	A2	A2	A2	A1	A2

(*) *Common European Framework of Reference (CEF) level*

Social skills and competences

- very good communication skills with people of different nationalities obtained during the Erasmus project;
- very good communication skills with students obtained while teaching university laboratories;
- good communication skills with teenagers obtained as high school and university prom organiser ;
- good communication skills with persons that have a handicap obtained as a volunteer;

Organisational skills and competences

- good organisational skills gained as B.Sc. co-advisor at the Faculty of Automation Computers & Electronics ,University of Craiova, Romania;
- good organisational skills gained as organiser of workshops for the Romanian Asociaction for Artificial Intelligence (ARIA);
- good organisational skills gained as student organiser of a computer science conferences;
- good team-leading skills obtained as prom organiser;

Computer skills and competences

- Operating Systems: Mac OS X, Windows, Linux/Unix;
- Programming languages: Python, C/C++, Turbo PASCAL, R, AutoIT, PROLOG;
- Data Base systems: MySql, Oracle, Neo4j;
- Version Control Systems: Git;
- Office suites: Microsoft Office, IWork, LaTeX;
- Other programs: Gephi, Weka, JFLAP;

Driving licence(s)

- B category

Academic information

Active member

- Intelligent Distributed Systems research group (ids.software.ucv.ro).
- Romanian Association For Artificial Intelligence (aria-romania.org).

Member of research grants

- Octomber 2015 - September 2017 ,"Platform for colobaborative learning based on social media instruments", Romanian Ministry of Education and Research, PN-II-RU-TE-2014-4-2604, director Conf.univ.dr.ing. Elvira Popescu.

Personal Grants

- Octomber 2013 - September 2016, Ph.D research grant, subject: Complex Networks, domain: Computers and Information Technology, doctoral advisor: Prof.univ.dr.ing. Costin Badica, University of Craiova, Romania.
- April 2014 - Octomber 2015, Ph.D research grant, supported by the strategic grant POS-DRU/159/1.5/2/133255. Project ID 133225(2014), cofinanced by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007-2013.
- October 2017 - March 2019, QFORIT Ph.D research grant on *Developping an adaptic negotiation protocol based on Complex Networks' theory*

B.Sc. co-advisor

- Janina Ninetta Bratu, *Mobile Recommender System for Entertainment and Leisure based on Weather and Location*,Faculty of Automation Computers & Electronics, University of Craiova, Romania, 2015.
- Petre Mazilu, *Analysis of opinion trends from unstructured data with aplicability in online tourism*, Faculty of Automation Computers & Electronics, University of Craiova, Romania, 2016.

M.Sc. co-advisor

- Andreea Calota, *Development of an augmented visualisation application for investigating students' colaboration in an e-learning platform*, Faculty of Automation Computers & Electronics, University of Craiova, Romania, 2017.

Honors and Awards

- *First prize*, in the 3rd Scientific Communication Session, organised by the Computer Science Department, Faculty of Sciences, University of Craiova, 2011.
- *Most Valuable Project* In the Category „Innovation", International Conference on Informatics Imagination, Creativity, Design, Development ICDD, Sibiu, Romania, April 26-28 2012.
- *Mention*, „STUDENT VISION" contest, organised by the Faculty of Sciences, within the Technical University of Cluj-Napoca, June 20th 2012.

Co-organiser

- *National (Romanian) Symposium of Scientifical Communication*, Department of Informatics, Faculty of Sciences, University of Craiova, Romania, 2012
- *7'th Balkan Conference in Informatics*, Sinaia, Romania, 2-4 September, 2015
- *Young Researchers in Artificial Intelligence 2015 (Romanian: Tineri Cercetatori în Inteligența Artificială)* Workshop, Cluj-Napoca, Romania, 5-8 September, 2015
- *High School Programming Contest (Romanian: Concurs de programare pentru licee ProCon) - Constantin Belea*, Department of Computers and Information Technology, University of Craiova, Romania, May 2016
- *Young Researchers in Artificial Intelligence 2016 (Romanian: Tineri Cercetatori în Inteligența Artificială TiCIA)* Workshop, Sinaia, Romania, 1-3 August, 2016
- *2016 IEEE International Conference on INnovations in Intelligent SysTems and Applications*, Sinaia, Romania, 2-5 August, 2016

Publishing and participation

2012

Event/Journal: 2nd International Conference on Informatics Imagination, Creativity, Design, Development ICDD, Sibiu, Romania, 26-28 April, 2012
Role(s): Research paper author & Participant
Publication: *Optimization of Gear Changing using Simulated Annealing*

Event/Journal: Annals of the University of Craiova-Mathematics and Computer Science Series 39.2: 309-321, 2012
Role(s): Reserch paper author
Publication: *Determine the number of twitter's user clusters*

2013

Event/Journal: 3rd International Conference on Informatics Imagination, Creativity, Design, Development ICDD, Sibiu, Romania, 16-18 May, 2013
Role(s): Research paper author & Participant
Publication: *Machine learning in gear changing*

2014

Event/Journal: 4TH WORKSHOP ON APPLICATIONS OF SOFTWARE AGENTS - WASA 2014 Thessaloniki, Greece, June 2-4, 2014
Role(s): Research paper author & Participant
Publication(s): *Agile development methods through the eyes of Organisational Network Analysis*

Event/Journal: 14th Workshop on Software Engineering Education and Reverse Engineering, Sinaia-Romania, 24-30 August 2014
Role(s): Keynote speaker & Participant

Keynote(s)	*Network Science*
Event/Journal	The 7th International Conference on Knowledge Science, Engineering and Management KSEM 2014, Sibiu, Romania, 16–18 October 2014
Role(s)	Research paper author & Participant
Publication(s)	*Complex Networks' Analysis Using an Ontology-Based Approach: Initial Steps*

2015

Event/Journal	5th International Workshop on Social Network Analysis, Anacapri, Capri island, Italy, 29-30 April, 2015
Role(s)	Participant
Event/Journal	19th IEEE International Conference on Computer Supported Cooperative Work in Design, Calabria, Italy, 6-8 May, 2015
Role(s)	Research paper author & Participant
Publication(s)	*Organisational Network Analysis withtin the Academical World: Initial Stepts* and *Initial Evaluation of an Ontology for Transport Brokering*
Event/Journal	Workshop on Social Media and the Web of Linked Data at EUROLAN 2015 Summer School on Linguistic Linked Open Data 18 July 2015, Sibiu, Romania
Role(s)	Research paper author
Publication(s)	*Towards Social Data Analytics for Smart Tourism: A Network Science Perspective*
Event/Journal	15th Workshop on Software Engineering Education and Reverse Engineering, Bled-Slovenia, 24-30 August 2015
Role(s)	Keynote speaker & Participant
Keynote(s)	*Using Complex Networks to Unterstand Tourist Reviews*
Event/Journal	7'th Balkan Conference in Informatics, Sinaia, Romania, 2-4 September, 2015
Role(s)	Research paper author, Co-organiser & Participant
Publication(s)	*Tourist review analytics using complex networks* and *Communication and interaction in a multi-agent system devised for transport brokering*

2016

Event/Journal	NESUS Winter School & PhD Symposium 2016, Timisoara, Romania, 8-11 February, 2016
Role(s)	Research paper author & Participant
Publication	*A Framework for Knowledge Management using Complex Networks Methods*
Event/Journal	International Summer School on Web Science and Technology, Bilbao, Spain, 18-22 July, 2016
Role(s)	Participant
Event/Journal	The 16th IEEE International Conference on Advanced Learning Technologies - ICALT2016, Austin, Texas, USA, 25-28 July, 2016

Role(s)	Research paper author
Publication(s)	*Predicting Student Performance and Differences in Learning Styles based on Textual Complexity Indices applied on Blog and Microblog Posts. A Preliminary Study*
Event/Journal	2016 International Symposium on INnovations in Intelligent SysTems and Applications, Sinaia, Romania, 2-5 August 2016
Role(s)	Research paper author, Co-organiser & Participant
Publication(s)	*Polarity Shifting for Romanian Sentiment Classification*
Event/Journal	16th Workshop on Software Engineering Education and Reverse Engineering, Jahorina-Bosnia & Berzegovina, 22-27 August 2016
Role(s)	Keynote speaker & Participant
Keynote(s)	*Online Resources for Teaching Programming to First Year Students*
Event/Journal	11th European Conference On TECHNOLOGY Enhanced Learning
Role	Research paper author
Publication(s)	*Predicting Academic Performance Based on Students' Blog and Microblog Posts*
Event/Journal	18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing SYNASC'16, Timisoara, Romania, 24-27 September, 2016
Role(s)	Research paper author & Participant
Publication(s)	*Automanted Negotiation Framework for the Transport Logistics Service*

2017

Event/Journal	18th INTERNATIONAL CARPATHIAN CONTROL CONFERENCE ICCC'2017, Sinaia, Romania, 28-31 May, 2017
Role(s)	Research paper author & Participant
Publication(s)	*Design of a Conceptual Knowledge Extraction Framework for a SocialLearning Environment based on Social Network Analysis Methods*
Event/Journal	14th Workshop on Agents for Complex Systems in the framework of SYNASC 2017 19th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, Timisoara, Romania, 21-24 September, 2017
Role(s)	Research paper author
Publication(s)	*Integration of Jason Reinforcement Learning Agents into an Interactive Application*
Event/Journal	21st International Conference on System Theory, Control and Computing ICSTCC 2017, Sinaia, Romania, 19-21 October, 2017
Role(s)	Research paper author
Publication(s)	*Using Social Network Analysis to Investigate Students' Collaboration Patterns in Emuse Platform*

2018

Event/Journal	Smart Learning Environments, Springer
Role(s)	Research paper author

Publication(s)	*Analyzing students' collaboration patterns in a social learning environment using StudentViz platform*
Event/Journal	Challenges and Solutions in Smart Learning, Springer
Role(s)	Research paper author
Publication(s)	*StudentViz: A Tool for Visualizing Students' Collaborations in a Social Learning Environment*
2019	
Event/Journal	International Conference on Computational Science - ICCS 2019, Faro, Algarve, Portugal, 14-16 June, 2019
Role(s)	Research paper author
Publication(s)	*Augmenting Multi-Agent Negotiation in Interconnected Freight Transport Using Complex Networks Analysis*
Co-authors	Prof.univ.dr.ing. Costin Badica, University of Craiova, Romania Prof.univ.dr. Scott Crossley, Georgia State University, USA Prof.univ.dr. Stefan Trausan-Matu, Politehnica University of Bucharest, Romania Prof.univ.dr.ing. Elvira Popescu, University of Craiova, Romania Conf.univ.dr. Catalin Stoean, University of Craiova, Romania Conf.univ.dr. Mihaela Colhon, University of Craiova, Romania Conf.univ.dr.ing. Mihai Dascalu, Politehnica University of Bucharest,Romania Asist.univ.drd.ing. Andrei Mocanu, University of Craiova, Romania Asist.univ.drd. Lucian Luncean, Romanian-German University of Sibiu, Romania

Additional information

Personal interests

Computer Science, History, Space Exploration, Foreign Policy

Prefered Romanian author

Neagu Djuvara

Prefered international author

Carl Sagan