UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Vitor Araújo Cautiero Horta

Detecting Semantic Overlapping Communities and Influential Nodes in Social Networks

Juiz de Fora

2018

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Vitor Araújo Cautiero Horta

Detecting Semantic Overlapping Communities and Influential Nodes in Social Networks

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Ciências Exatas da Universidade Federal de Juiz de Fora como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Orientador: Victor Ströele De Andrade Menezes
Coorientador: Jonice de Oliveira Sampaio

Juiz de Fora
2018

Ficha catalográfica elaborada através do programa de geração
automática da Biblioteca Universitária da UFJF,
com os dados fornecidos pelo(a) autor(a)

Horta, Vítor Araújo Cautiero.
Detecting Semantic Overlapping Communities and Influential Nodes in Social Networks / Vítor Araújo Cautiero Horta. -- 2018.
83 p. : il.

Orientador: Victor Ströele De Andrade Menezes
Coorientadora: Jonice de Oliveira Sampaio
Dissertação (mestrado acadêmico) - Universidade Federal de Juiz de Fora, ICE/Engenharia. Programa de Pós-Graduação em Ciência da Computação, 2018.

1. Social network analysis. 2. Overlapping community detection. 3. Clustering algorithm. 4. Ontology. 5. Semantic analysis. I. Menezes, Victor Ströele De Andrade, orient. II. Sampaio, Jonice de Oliveira, coorient. III. Título.

Vitor Araújo Cautiero Horta

Detecting Semantic Overlapping Communities and Influential Nodes in Social Networks

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Ciências Exatas da Universidade Federal de Juiz de Fora como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Aprovada em 22 de Novembro de 2018.

BANCA EXAMINADORA

Prof. D.Sc. Victor Ströele De Andrade Menezes - Orientador
Universidade Federal de Juiz de Fora

Prof. D.Sc. Jonice de Oliveira Sampaio- Coorientador
Universidade Federal do Rio de Janeiro

Prof. D.Sc. Regina Maria Maciel Braga Villela
Universidade Federal de Juiz de Fora

Prof. D.Sc. Geraldo Zimbrão da Silva
Universidade Federal do Rio de Janeiro

ACKNOWLEDGMENTS

A minha família, em especial meus pais Pedro Paulo e Márcia e irmãos André, Pedro e Bruno, por serem a base de todas as conquistas e pelo apoio incondicional.

A minha namorada Léia e família, que sempre acreditaram e valorizaram esta escolha, me passando toda a confiança necessária para alcançar os objetivos.

Aos amigos de longa data do clube 300, por estarem presentes em todos os momentos.

Ao Professor Victor pela sua participação essencial em minha formação acadêmica, me orientando em todos os passos neste caminho.

A Professora Jonice por sua colaboração essencial neste trabalho e por todos os conselhos importantes para a continuidade da minha carreira acadêmica.

Aos Professores José Maria e Regina, pelos diversos ensinamentos e conselhos que foram fundamentais e sempre serão levados adiante.

RESUMO

Comunidades em redes sociais são compostas por pessoas de interesses semelhantes que influenciam e são influenciadas pelo grupo. Identificar e explorar estas relações são fatores importantes que podem a apoiar a colaboração na rede. Neste estudo serão analisados os níveis de influencia entre pessoas e suas comunidades, considerando os aspectos estruturais das redes sociais e suas informações de contexto disponíveis. Para isso, conceitos de redes complexas e tecnologias de análise semântica são utilizadas para combinar análise estrutural e análise de conteúdo em redes sociais. Primeiramente é proposto um algoritmo para detecção de comunidades sobrepostas e pessoas imporantes em redes sociais, chamado NetSCAN. Como segunda proposta foi desenvolvida uma ontologia chamada NetO, cujo objetivo é apoiar análises semânticas em redes sociais. O algoritmo e a ontologia foram testados separadamente em experimentos controlados e em conjuntos de dados já conhecidos. Posteriormente, em duas avaliações de pesquisas históricas, o NetSCAN e a NetO foram utilizados em conjunto para realizar análise estrutural e semântica em duas redes sociais do mundo real. Na primeira avaliação uma rede social científica foi analisada, utilizando dados de um repositório científico chamado DBLP. A segunda avaliação analisou uma rede de desenvolvimento de software construída através de dados do StackOverflow, um dos mais populares fórums de perguntas e respostas (Q&A). A primera avaliação mostrou que foi possível detectar comunidades científicas, pesquisadores influentes e seus interesses de pesquisa. Na segunda avaliação, comunidades de desenvolvedores de software foram detectados, bem como desenvolvedores especialistas e seus tópicos de expertise. Os resultados apontam para a viabilidade e efetividade da solução proposta.

Palavras-chave: Social network analysis. Semantic analysis. Overlapping community detection. Clustering algorithm. Ontology.

ABSTRACT

Social network communities are composed of people with common interests who influence or are influenced by themselves. Identifying and exploring these relationships are fundamental activities to support collaboration in the network. In the present work we analyze the level of influence among people and their communities, by analyzing social networks considering both their structure and context information. To this, complex networks concepts and semantic technologies are used to combine linkage-based analysis with content-based analysis in social networks. We first propose an algorithm for detecting overlapping communities and important nodes in the network, named NetSCAN. Then, an ontology called NetO is proposed aiming to support semantic analysis. The proposed algorithm and ontology were first tested separately in controlled experiments based on well known datasets. Then, in two history research evaluations, they were used together to perform structural and semantic analysis over two real-world social networks. In first evaluation a Scientific Social Network was analyzed, based on a large scientific repository called DBLP. The second evaluation analyzed a Software Development Social Network constructed from StackOverflow data, which is one of the most popular question-answer (Q&A) forum in this context. The first evaluation has shown that we were able to detect scientific communities, influential researchers and their research interests. In the second evaluation, communities of software developers were detected as well as expert developers and their topics of expertise. Therefore, the results points to the viability and effectiveness of the proposed solution.

Keywords: Social network analysis. Semantic analysis. Overlapping community detection. Clustering algorithm. Ontology.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

Social networks have become very popular in recent years and many applications are using social network analysis to produce insight on data. According to Aggarwal (2011), a social network is defined as a network of *interactions* or *relationships*, where nodes consist of actors, and edges consist of relationships or interactions between these actors. The most popular examples are Online Friendship Networks such as *Facebook* and Twitter where the actors are people who have social interactions through an online platform.

However, the concept of social networks is not restricted to friendship network or internet-based systems, and it can be used to represent interactions in many different contexts. A classical example is the Milgram experiment (MILGRAM, 1967) that studied the *small world* phenomenon long before the advent of the internet. In the scientific context for example, Scientific Social Networks are used (STRöELE et al., 2016)(YANG et al., 2014) to represent the social relations established by researchers. Identifying and exploring these relationships are fundamental activities to support scientific experiments and collaboration. Other examples of networks can be found in the context of software development (HU et al., 2018), biology systems (KOSCHÜTZKI; SCHREIBER, 2008), traffic (JAYAWEERA et al., 2017) and many others (DAS et al., 2018).

The interdisciplinary nature of the subject have increasingly stimulated the study and development of algorithms and techniques to analyze network topology, define clusters for communities identification, and locate influencing elements, connectors and information diffusers.

Moreover, according to Cross and Parker (2004), there are four kinds of people in a social network, as follows: (i) central connectors, who have large amount of relationships; (ii) boundary spanners, who connect different groups of people; (iii) information brokers, influential people who communicate across subgroups maintaining a large connected group, or connecting two groups; and (iv) peripheral people, who are in the border of social network needing help to improve their connections. Analyzing all these kinds of people is important so as to characterize the social network in some way. Semantic meaning can also improve this characterization and specific analyses can be used to identify semantic connections among people who share similar interests that are not explicitly stated. In

addition, semantic meaning can be used to better characterize contexts and improve the connections among people related to these contexts.

Considering people in social networks, some studies focus on their direct connections (CROSS; PARKER, 2004; GUILLE, 2013), aiming to identify central connectors. However, indirect and implicit relationships are also important and should be considered in social network analysis (YANG; LESKOVEC, 2012; GRABOWICZ et al., 2012)(STRÖELE et al., 2018). In this sense, information brokers can influence people even if they are not directly connected. They help in the dissemination of certain kinds of information and in the connectivity throughout a network (CROSS; PARKER, 2004). Semantic characterization can also improve this process, identifying semantic interests and helping communities to maintain collaboration. In this regard, some studies have proposed methods to extract semantic knowledge in social networks (MENG et al., 2014) and methods for detecting communities in networks based on relationships enriched with semantic context (KIANIAN et al., 2017). This semantic knowledge can be extracted by using keywords of documents such as tags in forum posts (ERETEO et al., 2008). However, none of these studies have used implicit knowledge to discover new connections and propose new semantic contexts.

Thus, this study aims to find central connectors and information brokers in an attempt to identify influential people in social networks, connecting people that have similar interests, even though they are not explicit linked. For this, complex network concepts and techniques was used to analyze the interaction between nodes, identifying (i) influential people who work in two or more communities simultaneously, i.e., information brokers connecting two or more groups; and (ii) potential influencers in specific communities, i.e., central connectors from a group who connect subgroups. Ontological terms and rules was used to discover semantic meaning and, based on that, propose new connections between nodes, also including their contexts.

We defined two main research questions to guide the development of this work: **(i)** *How to identify nodes who help maintain the network connectivity, disseminating information and linking groups/subgroups?* **(ii)** *How to discover semantic connections between nodes, also including their connections based on their context, even though such connections are not explicit?*

In this vein, we first propose a novel algorithm for detecting overlapping communities

and influential nodes in bidirected social networks. Then, an ontology is also proposed to identify people interests and, by using ontological rules, identify new interest connections and contexts. Therefore, we are using complex networks to analyze people interactions and ontologies mostly focused on their semantic context.

As regards the contributions of this study, we can highlight: (i) the application of clustering techniques in large volume databases; (ii) the definition of network models for information brokers analysis aiming to identify influence among people based on bidirected graphs; (iii) the detection of communities and their subgroups; (iv) the identification of multidisciplinary people and their different influence levels, (v) the extraction of semantic information from the network and the use of this information to identify new connections and contexts, and (vi) the use of context-aware data, extracted from heterogeneous repositories, to deliver strategic information to communities and nodes. (vii) the combination of linkage-based and content-based analyses.

This study is organized as follows: Chapter 2 introduces the community detection problem and some related works; Chapter 3 describes the proposed NetSCAN algorithm for detecting overlapping communities and influential nodes; Chapter 4 describes the proposed NetO ontology for semantic analysis in social networks; Chapter 5 presents an evaluation in a real-world scientific social network; Chapter 6 presents another evaluation in a real-world software development social network and finally, Chapter 7 makes the final considerations.

2 COMMUNITIES IN SOCIAL NETWORKS

An important characteristic of a social network is its community structure. As the network evolves people tend to form groups, also named communities, which are sets of nodes with more and/or better interactions among its members than between its members and the remainder of the network (LESKOVEC et al., 2010). Finding these communities regards to the community detection problem and it has been drawing the attention of many researchers. There are many applications regarding this issue in different fields, such as viral marketing (YUAN et al., 2010), expert finding (WU et al., 2004), knowledge sharing (LIU et al., 2011) and others.

Several algorithms for detecting overlapping communities in social networks have already been developed with the aim of identifying groups whose members have greater similarity among themselves and greater dissimilarity from the members of other groups (HAN et al., 2011). In this context, a tertiary study was conducted to find the methods most commonly used to detect overlapping communities in social networks. The aim of that study was to find out secondary studies that in turn could help reveal the state of the art of a research area. Accordingly, five results were obtained (FORTUNATO, 2010; Puig-Centelles, Anna and Ripolles, Oscar and Chover, Miguel, 2008; WANG WEN-ZHONG TANG; WANG, 2015; XIE et al., 2013; PAPADOPOULOS et al., 2012). These articles presented methods with different approaches. Some examples are partitioning (HLAOUI; WANG, 2004), hierarchical (NEWMAN; GIRVAN, 2004; LUO et al., 2011) and density-based (FALKOWSKI et al., 2007; BHAT; ABULAISH, 2012) methods.

Partitioning methods aim to separate the data into k groups, minimizing some function of similarity between the points and the centers of these groups. In spite of the popularity of these methods, their main limitations are the high computational cost and the necessity of prior determination of the number of k groups.

Newman and Girvan (2004) and Luo et al. (2011) use divisive hierarchical methods for detecting communities. Basically, these methods consist of removing the edges with the greatest centrality (*edge betweenness*) and identifying communities in a dendrogram. This approach is frequently used but the high computational complexity of the method does not favor its use in large social networks.

DENGRAPH (FALKOWSKI et al., 2007) was developed using a density clustering strategy through an implementation of DBSCAN (ESTER et al., 1996), adapted for undirected graphs. As a result, it achieves better performance and greater ability to detect *noise* in networks with greater data volume. An approach for predicting dengue cases was also developed using the DBSCAN algorithm to estimate the incidence of dengue cases in Brazilian cities from dengue-related messages collected from social networks (STRöELE et al., 2016).

Since DBSCAN, in its original implementation, does not predict that the dataset may have multiple density granularities, (GIALAMPOUKIDIS et al., 2016) developed DBSCAN*-Martingale. This algorithm adapts the density parameters in order to discover group members with different similarity levels. However, an iterative process becomes costly when applied to a large data set. Li et al. (2016) observed the need to consider the direction in members' relationships in social networks. Consequently, they proposed a method that uses a density clustering strategy to detect communities in directed graphs. However, they do not consider overlapping communities, i.e, the possibility that one element belongs to two or more distinct groups.

The clique percolation method (DERÉNYI et al., 2005) (CPM) was one of the first methods proposed to detect overlapping communities. CPM strategy is to detect communities by finding adjacent *k-cliques* in the graph. Two *k-cliques* are considered as adjacent if they share *k-1* nodes. The adjacent cliques represent communities and the nodes, which belong to multiple adjacent cliques, representing overlaps between these communities.

Cminer is a density-based algorithm that, among the others described above, is the most similar to the one proposed in this study. The algorithm proposed by Bhat and Abulaish (2012) is based on DBSCAN and is designed for directed and weighted graphs. However, Cminer uses directed edges to calculate undirected distances between two nodes. In this case, the distances between two nodes A and B becomes undirected and $dist(a, b) = dist(b, a)$. This is the biggest difference between Cminer and the present study, since the method proposed here considers each directed edge separately.

Considering the semantic context, some approaches have been proposed in the literature. In (ERETEO et al., 2008), the authors review concepts related to social network analysis and semantic web. The authors show that semantic context in social networks can be analyzed through keywords used by individuals, such as tags in forum posts. Another

proposal that uses tags to extract semantic context in networks is FuSeO (KIANIAN et al., 2017). FuSeO is an approach for the detection of overlapping communities focused on Q&A (questions-answers) forums that extract topics of interest among users using their tags in forums. By using fuzzy relations, FuSeO is able to find the users' interest level considering the topics. Later, FuSeO proposes the use of an algorithm to detect overlapping communities. Despite the use of an ontology in their approach, the authors mention that this is a lightweight ontology, without the ability to derive implicit knowledge. Another proposal to extract semantic context through topics of interest in forums is QASM (MENG et al., 2014). QASM aims to detect topics of interest and the users' expertise level in each topic. To determine these topics, QASM uses a probabilistic model. This is one of the main differences between the semantic analysis of this study and QASM proposal, since our study uses an ontology and a domain taxonomy for the discovery of topics of interest.

The first distinguishing feature of this study is the development of a clustering algorithm that considers different levels of influence among pairs of individuals (bidirected graph). Secondly, in terms of the semantic analysis our study proposes an ontology that is capable of extracting implicit knowledge by using inference and logical rules.

Therefore, from the point of view of structural analysis, this work advances in the state of the art by considering each directed edge separately in bidirected relationships and by detecting overlapping communities with a non-iterative method suitable for execution in large complex networks. In the semantic context we propose an ontology for detecting nodes's interests with the capability of extracting implicit knowledge, being it also a step forward compared to the previous works. The integration of the proposed community detection algorithm and the ontology is a novel solution to combine content-based analysis with linkage-based analysis.

3 NETSCAN ALGORITHM

The first main propose of this work is an algorithm for tackling the community detection problem. NetSCAN (HORTA et al., 2018b) is a density-based method for detecting overlapping communities and influential nodes in social networks. Similarly to DBSCAN (ESTER et al., 1996) the algorithm searches for *core* nodes which are nodes with a dense neighborhood. When a code node is found it aggregates all its neighbors in a community. If any node in this neighborhood is also a *core* node it expands the community by including its own neighbors.

One of the main differences with respect to DBSCAN is that NetSCAN considers that the distance between two vertices depends on the direction of the focused relationship. In addition, it allows the same vertex to be included in more than one group and, in this way, it is possible to identify people who contribute in different areas and in distinct groups, even if they are not considered as centralizers.

NetSCAN has three parameters, two mandatory ones (*eps* and *minPts*) with characteristics similar to those of DBSCAN parameters, and an optional one, which allows the searching for elements influenced by the *core* in a greater depth (*radius*). So, it allows defining how many layers of neighbors from the *core* will be analyzed, as shown in Figure 3.1, where the maximum value of the parameter radius is equal to the network diameter given by n.

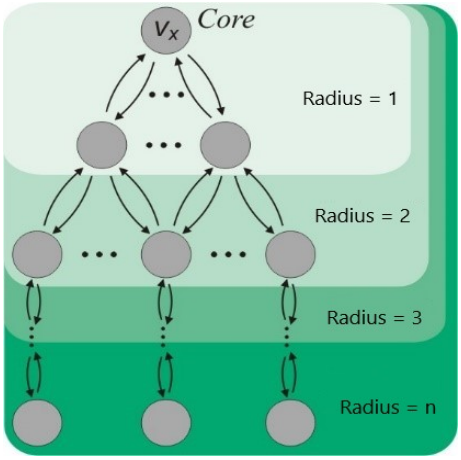


Figure 3.1: Illustration of the parameter "radius"

Parameter *eps* defines the minimum influence for a vertex to be considered an influencer of its neighbor, and *minPts* indicates the minimum number of neighbors that a

vertex should have to be considered a core. The pseudocode of the algorithm is described in the algorithm in Figure 3.2.

Algorithm 2 Network Scan

```
1: function NETSCAN(GRAPH,EPS,MINPTS,RADIUS)
2:     clusterId ← 0
3:     while N = getUnclassifiedNode(graph) do
4:         clusterId ← clusterId + 1
5:         neighbors ← localSearch(N, grafo, eps, raio)
6:         expandNode(N, neighbors, graph, eps, minPts, clusterId, radius)
7:     end while
8: end function

1: function EXPANDNODE(N,NEIGBORS,GRAPH,EPS,MINPTS,CLUSTERID,RADIUS)
2:     numNeighbors ← length(neighbors)
3:     if (numNeighbors <minPts) then
4:         setNoise(N)
5:         returnfalse
6:     end if
7:     setCore(N)
8:     groupNeighbors(N, neighbors, graph, eps, minPts, clusterId, radius)
9: end function

1: function GROUPNEIGHBORS(N,NEIGBORS,GRAPH,EPS,MINPTS,CLUSTERID,RADIUS)
2:     group(N, neigbors, clusterId)
3:     for i ← 1 until length(neighbors) do
4:         N ← neighbors[i]
5:         newNeighbors ← localSearch(N, graph, eps, radius)
6:         expandNode(N, newNeighbors, graph, eps, minPts, clusterId, radius)
7:     end for
8: end function
```

Figure 3.2: NetSCAN algorithm

The first step of the algorithm is to select a v_x vertex, randomly, that has not been visited before. Afterwards, a region search is performed and a set with all neighbors of v_x is returned. This set represents all vertices that are influenced by v_x with an influence greater than or equal to *eps*. If the number of vertices returned in the search is greater than or equal to *minPts*, a group G is formed, the element v_x is defined as a *core* that will be further expanded. Expanding v_x element means grouping its neighbors in G, and then looking for any *core* among its neighbors. If any neighbor of v_x is also a *core*, its neighbors will also be parsed and grouped in G.

Nodes that do not have a number of neighbors greater than or equal to *minPts* will be marked as a noise. If they are close neighbors, and consequently grouped by some *core*, they can be considered as *border points*, being linked to the group of their *core*. If the optional *radius* parameter, which has a default value of 1, is modified, the *Region Search* function will include in its search the vertices that are more than one distance edge from the node being analyzed, that is, it will search beyond the layer of *radius* equal to 1 (Fig. 6). In this case, if v_x is a *core*, v_y is a *border point* influenced by v_x , and v_z is a *noise* influenced by v_y. For a *radius* greater than or equal to 2, the region search will consider

that v_z is a neighbor influenced by v_x and therefore will be grouped becoming a *border point* linked to the *core* group v_x . The use of this parameter allows a node, influenced by another, to be linked to the central influencer group, enabling a more detailed analysis of the influence levels of the nodes. For performance reasons, and to ensure that the grouping process can be stopped and resumed without loss of information, *"LocalSearch"*, *"Group"*, *"SetCore"* and *"SetNoise"* functions were implemented through executable queries in the Neo4j database.

3.1 NETSCAN IN TEST

As previously defined, this study intends to identify communities and important people that could help maintain the network connectivity, disseminating information and linking groups/subgroups.

Considering the amplitude of the solution, this section presents an evaluation of NetSCAN through three benchmark datasets. The objective is to test only NetSCAN behavior and its ability. In this preliminarily test, semantic behavior is not evaluated, since the datasets do not have semantic information that can be used to process semantic analysis. Three experiments were carried out to test NetSCAN in well-known databases:

- An experiment to evaluate the algorithm regarding the correctness of the groups, in which synthetic data were used;
- An experiment using synthetic social networks to identify overlapping communities;
- An experiment in small real-world social networks to identify overlapping communities and to compare the results obtained by NetSCAN with the results of the literature;

3.1.1 SYNTHETIC DATA TEST

The aim of the test using synthetic data is to evaluate the ability of the algorithm to detect clusters in classical cases of clustering problem, in which the connected vertices are equidistant from each other and there is no notion of direction in relationships.

This test was carried out on three different datasets of synthetic data with the purpose of verifying the algorithm behavior in simple and well-behaved databases. Thus, these

datasets were created in R^2 to make it possible to perform a controlled experiment and visualize the NetSCAN results. The visualization allowed analyzing whether the identified groups were well formed according to the data structure. For this experiment, we used three datasets (http://labic.ic.uff.br/Instance/):

- **200Data**: it is a dataset with 200 points in a normal distribution in R^2. All these points are well distributed in the space, as can be seen in Figure 3.3.
- **2Face**: it is a dataset generated to analyze density-based algorithms in R^2. The points in the dataset form a face-shaped drawing combined with points creating number 2 (Figure 3.4).
- **Numbers**: it is a dataset where the points draw numbers from 0 to 9 in the R^2 space. This dataset can be seen in Figure 3.5.

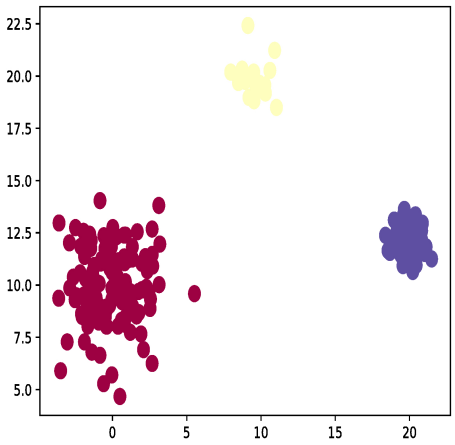


Figure 3.3: Result of grouping for the 200Data instance.

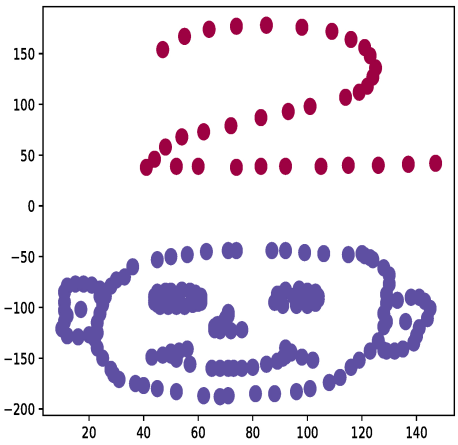


Figure 3.4: Result of grouping for the 2Face instance.

Figure 3.5: Result of grouping for the Numbers instance.

NetSCAN was implemented to consider directed relationships and, therefore, in this test, the weights of the edges were defined from the Euclidean distance and all the vertices have connections to and from each other.

In this test, a visual analysis of the results was performed, as it is a controlled evaluation in which all data have two dimensions (R^2). Thus, Figure 3.4 shows that the proposed algorithm recognized the two distinct groups in the *2Face* instance: the points that form the face and the points drawing number 2. In Figures 3.3 and 3.5, we can see the effective detection of the groups in the *200Data* dataset representing groups with different density, and the numbers defined in the *Numbers* dataset. As the algorithm is based on density, these databases were used to evaluate the correctness of the groups identified by NetSCAN.

The obtained results showed that the algorithm can identify groups correctly. However, there are no guarantees regarding the algorithm's functioning in the detection of overlapping communities, in social networks datasets. The following sections address social networks features to evaluate NetSCAN from other perspectives.

3.1.2 SYNTHETIC SOCIAL NETWORK TEST

Despite of the satisfactory results previously obtained, such instances are not able to reflect all the structural characteristics of a social network, such as asymmetric relationships among individuals. Therefore, to observe the behavior of NetSCAN in a scenario with characteristics of a social network, tests were conducted on a synthetic network and two real networks. This evaluation aims to verify the effectiveness of NetSCAN in identifying (i) social network communities, and (ii) people who are information brokers.

For this test, it was necessary to define another measure for the weight of the edges, since the relationships in these networks at first do not have a definition of distance and are not directed either. For this purpose, the weight measure applied was based on the influence metric, as defined in Equation 3.1.

$$IP(e_{ij}) = \frac{\parallel P(V_i) \bigcap P(V_j) \parallel}{\parallel P(V_i) \parallel} \tag{3.1}$$

In this equation, $IP(e_{ij})$ is the weight of edge e_{ij}, $P(v_x)$ is the set of all interactions of node v_x , and $|\cdot|$ is the number of elements in the set.

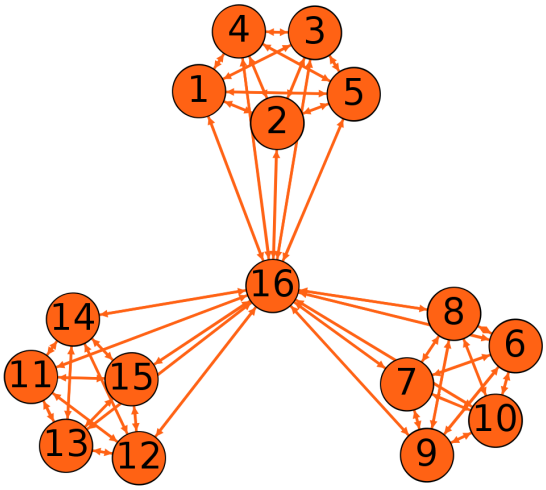


Figure 3.6: Artificial social network devised for the experiment.

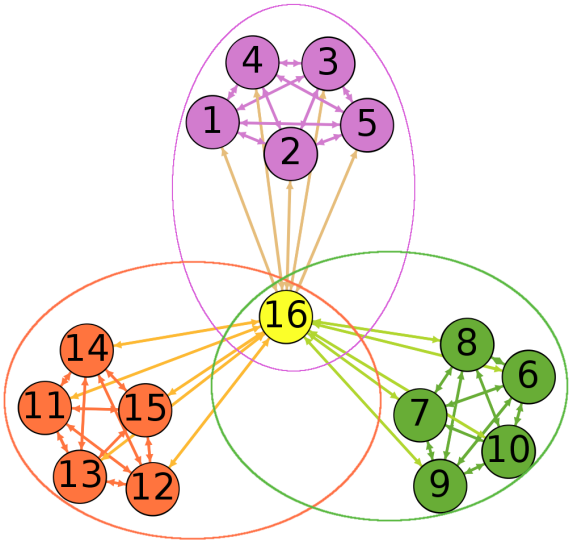


Figure 3.7: Result of the grouping carried out on the artificial network.

The artificial network in Figure 3.6 was devised to have 3 well-defined communities that overlap through a central vertex. It is important to highlight the difficulty in assigning a correct group to the central vertex of number 16, since it interacts with the 3 communities at the same intensity. As shown in Figure 3.7, NetSCAN detected the 3 communities and considered that there is an overlap between these groups. As a result, it made vertex 16 (yellow), which participates in all groups simultaneously, characterizing it as an information broker, reaching the expected result for this instance. The result for this instance is exactly the same as that found in the literature (MEENA; DEVI, 2015), which uses a genetic algorithm to detect communities. Hence it is considered to be a correct result.

Although NetSCAN showed satisfactory results with data that simulate a social network, other tests were performed to validate it in a real context, as shown in the next section.

3.1.3 REAL-WORLD CONTEXT DATA

In this third test, we aimed to evaluate the behavior of the algorithm in real-world networks. We used the data available at UCI Network Data Repository (`https://networkdata.ics.uci.edu/index.php`.), which is a well-known dataset repository for scientific study of networks. Two small real-world social networks were used, enabling us

to compare the NetSCAN results with the results obtained by other clustering approaches. The first one is the Zachary's Karate Club (ZACHARY, 1976) and the second dataset is the Protein Network (MEENA; DEVI, 2015). These are small datasets, so a visual analysis of the results obtained by NetSCAN is possible.

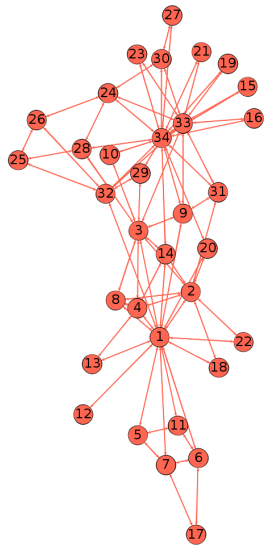


Figure 3.8: Karate community.

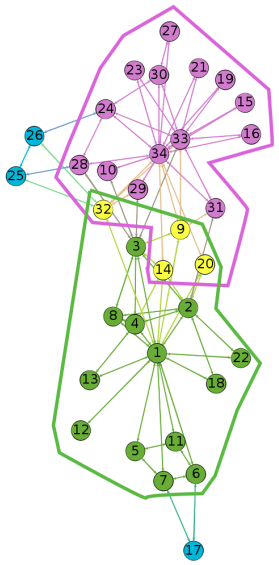


Figure 3.9: Result of the clustering formed in the karate network.

The first one (Figure 3.8) is a social network known as the Zachary's Karate Club Network. It shows the social relationships among the 34 members of the karate club. In this instance, there are 34 nodes, 78 edges and 2 real communities. As shown in Figure 3.9, the algorithm detected two existing communities, considering 3 individuals as outliers (blue) and 4 individuals that participate in both communities at the same time (yellow), characterizing an overlap between these groups. We compared these results with six other results achieved by different methods presented in (BHAT; ABULAISH, 2012) and (MEENA; DEVI, 2015). The compared methods were: density-based methods, modularity optimizers, genetic algorithms and clique-percolation based method. In this comparison, it was found that NetSCAN is the only method that was capable to find the only two known real communities. Although it is difficult to confirm that this is the most correct result, it can be seen that NetSCAN results are viable for this instance.

Figure 3.10 illustrates a real-world network containing 21 protein vertices, 61 edges, three different groups and two overlapping vertices. As shown in Figure 3.11, NetSCAN again found three communities and correctly identified vertices 8 and 9 (yellow) as overlapping. This is the same result found in the literature (MEENA; DEVI, 2015), so it was

considered that NetSCAN achieved the correct results for this instance.

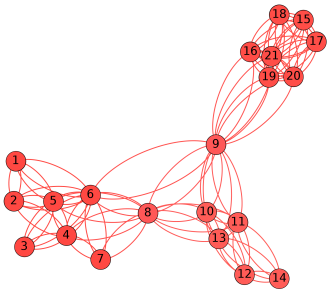


Figure 3.10: Protein network with overlaps in 2 vertices.

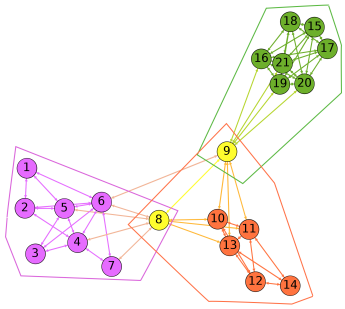


Figure 3.11: Result of the clustering formed in the protein network.

This benchmark shows that NetSCAN achieves the same results as the state of the art methods for two datasets (synthetic network and PPI network) and for the Karate Club network the NetSCAN performs even better than the state of the art methods by being the only algorithm to find the two real communities in the network (ZACHARY, 1976). In the case of Karate network, we argue that NetSCAN performs better than other density-based methods because it considers the bidirected relationships instead of modifying the original structure by unifying the relationships.

Based on these tests, we can affirm that NetSCAN obtained satisfactory results both in identifying social network communities and in identifying the overlaps between them. However, although used for tests, datasets without bidirected relationships may not justify the use of NetSCAN since other methods in this case can achieve the same results.

Based on these three preliminary evaluations, we know that NetSCAN is able to identify communities and elements involved in more than one community. However, these evaluations are specific to test NetSCAN behavior and were performed in small benchmark datasets. In order to evaluate the overall proposal, Chapter 5 and Chapter 6 presents experiments in large-sized real-world social networks in order to analyze the real communities and their overlaps, using structural and semantic analysis. In next chapter the NetO ontology is presented for support semantic analysis in social networks.

4 NETO ONTOLOGY

To support semantic analysis in social networks we propose an ontology, named NetO. The objective of NetO ontology is to identify the semantic context of nodes and extract topics of interest for each actor in the network. The ontology was created with Protégé[1] and makes use of Classes, Data Properties, Object Properties and ontological rules. Its class hierarchy is shown in Figure 4.1 and object properties are shown in Figure 4.2.

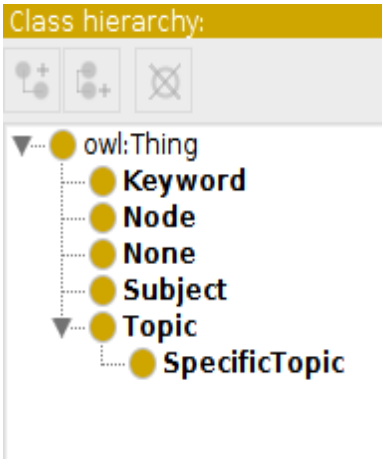


Figure 4.1: NetO class hierarchy.

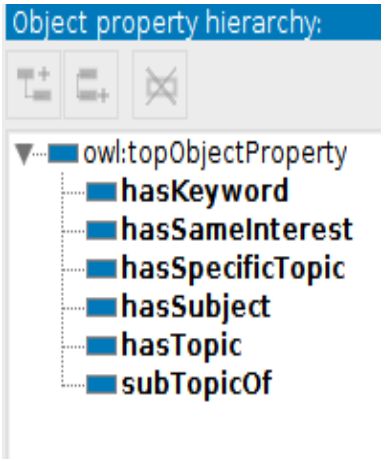


Figure 4.2: NetO object properties.

Class hierarchy includes five classes: **Node**, **Keyword**, **Subject**, **Topic** and **Specific Topic**. Table 4.1 summarizes the role of each class and their relations through object properties.

Table 4.1: NetO class details

Class	Represents	Instatiated by	Related to
Node	Individuals	Automatic extraction process	Keyword through "hasKeyword"
Keyword	Tags of social objects	Automatic extraction process	Subject through "hasTopic"
Subject	Relevant concepts	Predefined in the ontology	Topic through "hasTopic"
Topic	Topic of interest	Predefined in the ontology	SpecificTopic through "hasSpecificTopic"
Specific Topic	Specific topic of interest	Predefined in the ontology	-

[1]https://protege.stanford.edu/

Node class is used to represent individuals in the network. **Keyword** class represents keywords that can be found in the semantic context of these individuals. **Subject** class represents subjects addressed by each individual and it can be derived from common knowledge bases such as domain taxonomies. **Topic** class refers to topics of interest, and their individuals are inferred from the **Subject** class. Some topics may be related to each other and it might be interesting to be represented when a node have interest in pairs of related topics. In NetO these case is represented by the **SpecificTopic** class.

Individuals from **Subject**, **Topic** and **SpecificTopic** classes represents domain knowledge of the social network context and can be defined based on knowledge representations such as taxonomies, other ontologies or commonsense knowledge. Once the domain knowledge is specified, individuals from **Node** and **Keyword** classes can be automatically extracted from the data source.

To extract topics of interest of each **Node** class individual, the following ontological rules were also specified in SWRL (Semantic Web Rule Language) (https://www.w3.org/Submission/SWRL).

Rule1: Keyword(?k) ∧ Subject(?s) ∧ name(?k, ?n1) ∧ name(?s, ?n2) ∧ swrlb:equal(?n1, ?n2) → hasSubject(?k, ?s)

Rule2: Node(?n) ∧ hasKeyword(?n,?k) ∧ hasSubject(?k,?s) → hasSubject(?n,?s)

Rule3: subTopicOf(?s1, ?s2) ∧ subTopicOf(?s2, ?s3) → subTopicOf(?s1, ?s3)

Rule4: Node(?n) ∧ hasSubject(?r, ?s1) ∧ subTopicOf(?s1, ?s2) ∧ Subject(?s2) → hasSubject(?n, ?s2)

Rule5: Node(?n) ∧ hasSubject(?r, ?s) ∧ Topic(?s) → hasTopic(?n, ?s)

Rule6: Node(?n) ∧ hasTopic(?n, ?t1) ∧ hasTopic(?n, ?t2) ∧ hasSpecificTopic(?t1, ?st) ∧ hasSpecificTopic(?t2, ?st) ∧ differentFrom(?t1, ?t2) → hasSpecificTopic(?n, ?st)

Rule7: Rule7: Node(?p) ˆ Node(?p2) ˆ Topic(?t) ˆ hasTopic(?p,?t) ˆ hasTopic(?p2,?t) → hasSameInterest(?p, ?p2)

The steps for performing semantic analysis with NetO support are described as follows:

- Step 1 - Preprocessing: Identification of a taxonomy with terms related to the network domain.

- Step 2 - Preprocessing: Extraction of keywords related to nodes of the network and mapping of these keywords in terms of the domain taxonomy.
- Step 3 - Instantiation: Instantiate the network nodes and extract keywords in NetO ontology.
- Step 4 - Extraction: Inference machine processing for extraction of topics of interest.
- Step 5 - Extraction: Inference machine processing to discover new connections between nodes based on their topics of interest.

To illustrate the semantic analysis processing we have elaborated an example in a scientific network context. In this example, consider that nodes are researchers in computer science area and that the keywords can be extracted from their publications. The subjects can then be defined based on a computer science taxonomy, such as Computing Classification System (CCS) (`https://dl.acm.org/ccs/`), developed by ACM.

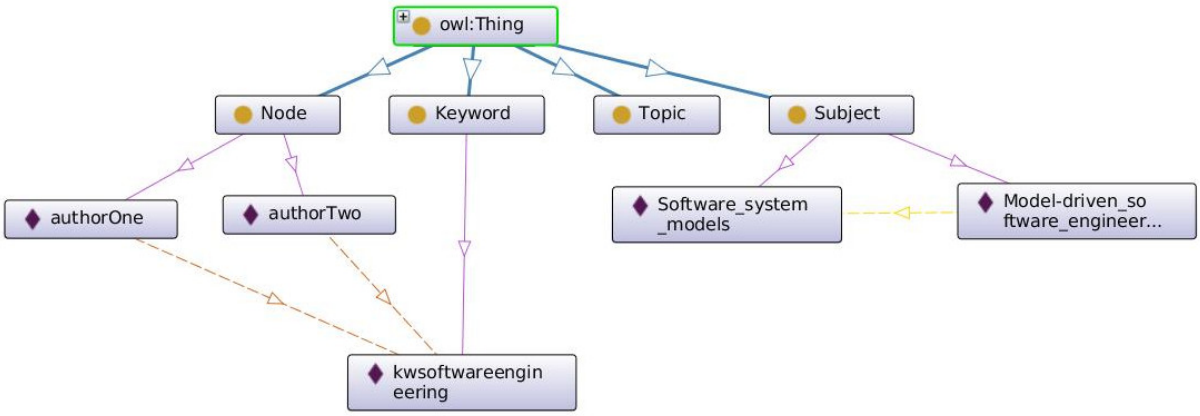


Figure 4.3: Individuals shown in a graph representation.

After preprocessing and instantiation of individuals in the ontology, the last two steps include the use of inference machine to extract topics of interest and discover new connections between individuals.

Thereby, as shown in Table 4.2 with DL (Description Logic) and in Figure 4.3 with graph representation, *authorOne* and *authorTwo* represent two individuals of **Node** class who used **Keyword** *kwsoftwareengineering* in one of their publications. The **Keyword** *kwsoftwareengineering* has the name "softwareengineering". The **Subject** *Model-driven software engineering* was preprocessed from the CCS ontology and has two names: "softwareengineering" and "software-engineering". In addition, the **Subject** *Model-driven soft-*

ware engineering was defined as the *subtopic* of **Subject** *Software system models*, which, in turn, does not have a *subtopic*.

Table 4.2: Individuals described in DL.

Description logic
authorOne authorOne : Person hasKeyword(authorOne, kwsoftwareengineering)
authorTwo authorTwo : Person hasKeyword(authorTwo, kwsoftwareengineering)
kwsoftwareengineering kwsoftwareengineering : Keyword name(kwsoftwareengineering ”softwareengineering”)
Model-driven software engineering Model-driven software engineering : Subject subTopicOf(Model-driven software engineering, Software system models) name (Model-driven software engineering ”softwareengineering”)
Software system models Software system models : Subject subTopicOf(Software system models, None)

Thus, in addition to enriching the results obtained by NetSCAN, semantic analysis allows examining the network context, helping to discover the real reasons of community formation and the identification of influential individuals in specific areas.

After the inference machine execution and using the SWRL rules presented above, the NetO ontology associated, based on the **Keyword** *kwsoftwareengineering*, that *authorOne* and *authorTwo* have the **Subject** *Model-driven software engineering*. As *Model-driven software engineering* is a *subtopic* of *Software system models*, it is also inferred that *authorOne* and *authorTwo* have the **Subject** *Software system models*. Thus, *Software system models* is considered a *topic* and it is inferred that *authorOne* and *authorTwo* have the topic of interest, i.e., *software system models*. As *authorOne* and *authorTwo* have the same topic of interest, there is also a new connection between them, represented by the *hasSameInterest* relation. These results can be seen in Figure 4.4.

From the example above, it can be noticed that NetO ontology is able to identify semantic context of researchers (nodes in the example network) through the recognition of their topics of interest. Through “hasSameInterest” inferred relation, it is possible to enrich the network by strengthening existing interactions or by creating new connections

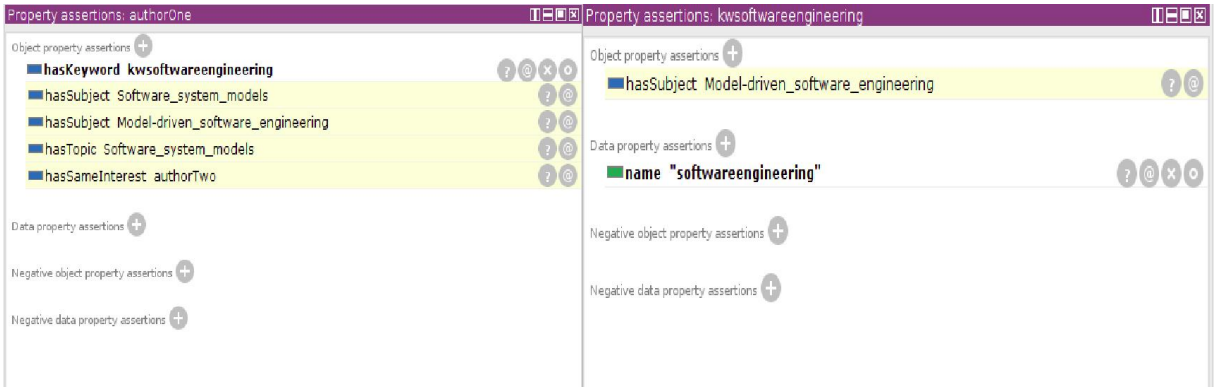


Figure 4.4: Inferred results achieved by the inference machine execution in the example.

between individuals.

Thus, in addition to enriching the results obtained by topological analysis, semantic analysis allows examining the network context, helping to discover real reasons of community formation and identification of influential individuals in specific areas. This processing can be performed before or after structural analysis in social network. If used before, network will be enriched with new connections and topics of interests and the structural analysis can take benefit from this. If used after, this process can complement the structural analysis by adding context knowledge to it.

Since all elements of the proposal were presented and tested in some way, the next two chapters evaluates the use of NetSCAN algorithm with NetO ontology on large-sized real-world social networks. The first study analyzes interactions between researchers in a large scientific social network (DBLP). We first perform topological analysis to characterize the network structure and to detect scientific communities and influential researchers. Then, a semantic analysis is used to extract semantic meaning and context knowledge on found communities and researchers.

The second study analyzes relationships between developers in a software development social network based on question-answer forum StackOverflow. In this study the semantic analysis is first used to enrich the social graph. Then, topological analysis is performed in the enriched graph to automatically detect software development communities, expert developers and their topics of interests.

Both studies use the History Research evaluation method combining linkage-based with content-based analysis to detect communities, influential nodes and their semantic meaning in these social networks.

5 REAL-WORLD CO-AUTHORSHIP SOCIAL NETWORK EVALUATION

After analyzing and testing the proposed algorithm operation and NetO ontology, in this section we analyze a real-world scientific social network in order to answer the previously stated research questions, identifying research communities and influential researchers, using structural and semantic analysis (central connectors and research-information brokers).

The scope of this evaluation was based on the GQM method (KITCHENHAM, 2007), described as follows: *"**To analyze** topologically and semantically the scientific social network, its overlapping scientific communities and scientific repositories **for the purpose** of providing information for decision-making concerning research-information brokers and central connectors **in relation to** scientific communities **under the point of view** of researchers and scientific communities **in the context** of heterogeneous distributed scientific repositories"*. Based on the scope definition, research questions were defined:

- (i) How to identify researchers who help maintain the network connectivity, disseminating information and linking research groups/subgroups?
  - RQ1: Are there researchers who work in two or more communities simultaneously, characterizing research-information brokers?
  - RQ2: Are there researchers who are largely connected to other researchers, characterizing central connectors?
  - RQ3: Does the use of cluster analysis help discover real scientific communities considering the activities developed by researchers?
- (ii) How to discover semantic connections (research interests) between researchers, also including their connections based on their scientific context, even though such connections are not explicit?
  - RQ4: Does the use of semantic analysis reinforce connections and propose new ones?

The first question helps finding whether the proposed solution is capable to detect influential researchers' and research communities with similar semantic context in the scientific network. The second question intends to explain whether it is possible to derive implicit knowledge from the scientific network and the different data sources.

According to Yin (2008) the History Research is the recommended evaluation method to answer explanatory questions when there is virtually no control over the events. The advantage of the historical method is that it does not depends on direct observations of the events. Instead, the study relies on documents and artifacts as the main sources of evidences.

Considering that we evaluated historical data from a real scientific social network and that we do not have control of researchers' behavioral events, we chose to use the History Research for the evaluation method. As sources of evidence, archival records were used, followed by a semantic analysis. The first data source was used to verify the identified groups in general, the second one was used to evaluate some communities and specific relationships, through a detailed analysis of the obtained results.

5.1 SCIENTIFIC SOCIAL NETWORK MODELING

Scientific Social Networks are specific types of social networks that represent the social interactions that occur in the scientific environment. For the development of this study, the scientific social network was constructed with the researchers' data extracted from the DBLP database, using its xml data file (`http://dblp.uni-trier.de/xml/`). Given the large volume of data and for a better understanding of the structure, the relational schema of this database was converted into a graph-oriented model, and the data were stored in a native graph database (Neo4j).

In the scientific network model used herein, nodes of the graph represent researchers, and the edges have weights that represent the co-authorship influence between them. This influence is calculated based on co-authorship relationships between researchers and helps identifying how much one researcher contributed for another researcher to stay active in the network. The influence one researcher exerts on another is not necessarily equal to the influence he/she receives from that same researcher. Therefore, the social graph representing this model is a bidirected graph $G = (V, E)$, where $V = \{v_0, v_1, ..., v_{n-1}\}$ is the set of n nodes (researchers) and E is the set of m edges $e_{ij} = v_i$, v_j between researchers

v_i and v_j , $0 \leq i < n$ and $0 \leq j < n$ (HORTA et al., 2017). The weights (roundtrip) of the relationship between two researchers are analyzed in a differentiated way, so the relationship represented by edge e_{ij} is, in general, different from that represented by edge e_{ji} , i.e., $e_{ij} \neq e_{ji}$.

Figure 5.1 illustrates an abstraction of the model proposed in this study where two vertices, v_i and v_j , are always connected by two directed edges, with independent weights. The weight of the edges is defined according to Equation 5.1, where $IP(e_{ij})$ is the weight of edge e_{ij} , representing the influence that researcher v_j has on researcher v_i, $P(v_x)$ is the set of all publications by researcher v_x, and $|\cdot|$ represents the number of elements in the set. It is worth noting that the weight of the edges vary in the range of 0 to 1. Values close to 0 indicate a weak influence showing that the majority of v_i publications are independent from v_j. On other hand, values close to 1 indicate a strong influence and shows that v_j contributions are important to keep v_i active in the network.

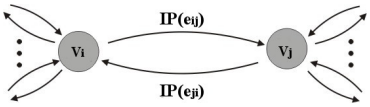


Figure 5.1: Relationships between two researchers.

$$IP(e_{ij}) = \frac{\parallel P(V_i) \bigcap P(V_j) \parallel}{\parallel P(V_i) \parallel} \tag{5.1}$$

In large-scale social networks, the analysis of their structure allows us to understand how the relationships between elements occur. So, based on this model, next section analyzes the topology of the co-authorship social network (based on DBLP) in order to characterize its structure and verify how researchers' relationships are distributed.

5.2 DBLP NETWORK TOPOLOGY

After the construction of the scientific social network based on DBLP, according to the proposed model, it is possible to analyze its topology. This analysis consisted of searching for non-trivial characteristics that facilitate the understanding of the network structure and studying how links between researchers are constructed. A challenge that impacts the manipulation and analysis of this network is the volume of data involved, since there are 1,306,546 vertices and 9,915,146 edges. In order to analyze the network structure, its degree distribution was calculated. This property, which characterizes the topology of a complex network, is obtained by calculating the number of nodes that have a certain

degree for all values of degrees present in the network. Degree distribution can be used to identify the type of network. In general, these networks can be classified as random, free of scale, modular, small world, among others (WASSERMAN; FAUST, 1994). Equation 5.2 was used to calculate this distribution, where $f(k)$ is the fraction of vertices that have degree equal to k.

$$f(k) = \frac{Number\ of\ nodes\ w\ ith\ K\ degree}{Total\ number\ of\ nodes} \tag{5.2}$$

In order to identify the type of scientific social network of this study, the degree distribution of its elements was calculated. Graphs in Figure 5.2 show the results obtained. As can be seen in these graphs, there are many researchers with few relationships (low degree) and few researchers with many relationships (high degree). This characteristic is typical of scale-free networks, in which few elements centralize most network relationships, complying with a power law.

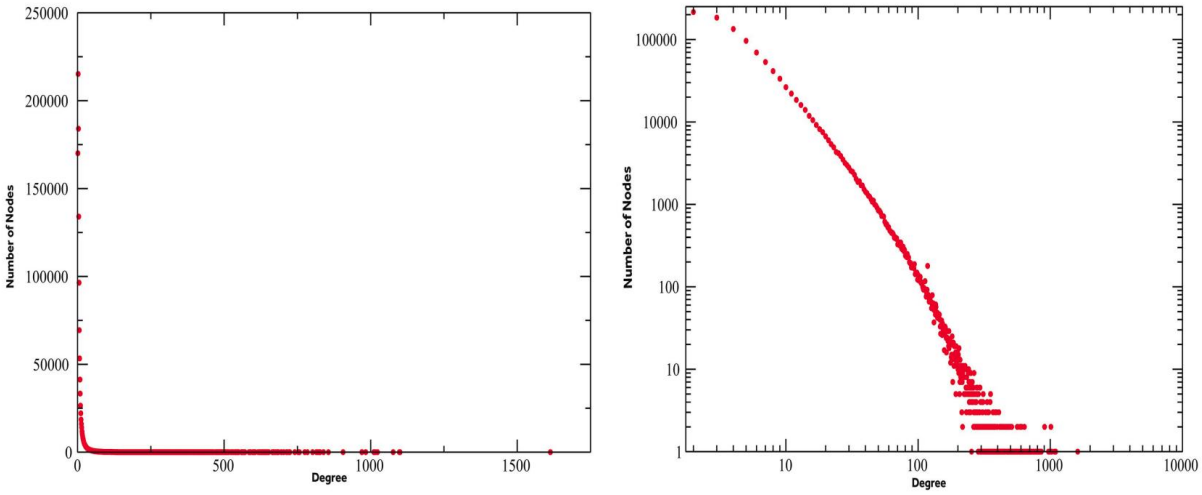


Figure 5.2: Degree distribution of Scientific Social Network graph from DBLP. Node degree distribution (left) and corresponding log–log graph (right).

Figure 5.3 displays the comparison between DBLP degree distribution and the respective adjusted power law curve.

Even though the curves have a similar behavior, there is a large distance between the DBLP degree distribution and the adjusted power law curve for some initial points in the graph. For this reason, it is not possible to conclude that the DBLP degree distribution follows a power law. However, because of the heavy-tailed degree distribution, this network has a preferential attachment and the most connected researchers are more likely to receive

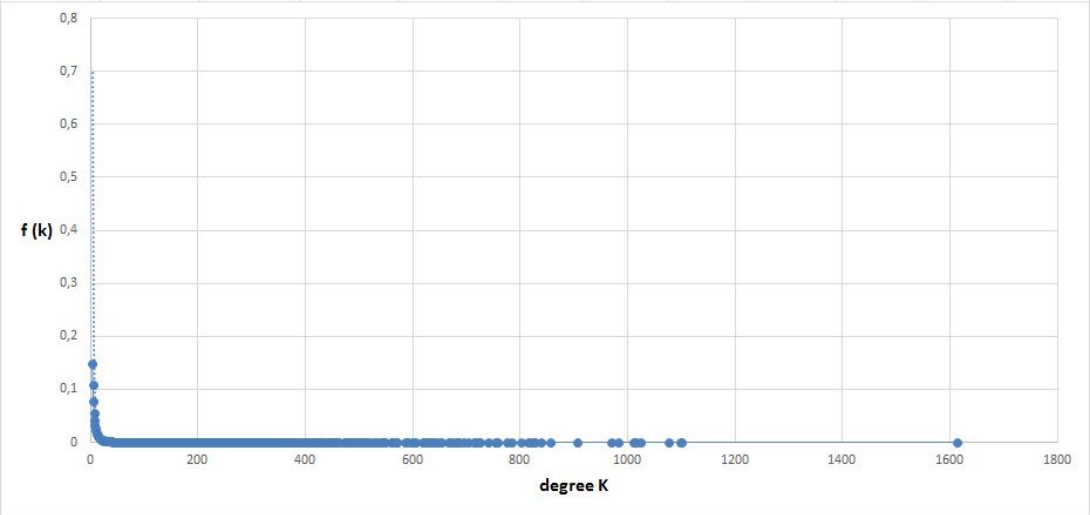


Figure 5.3: Degree distribution and adjusted power law curve.

new connections.

Based on the degree distribution behavior, it is possible to verify the coherence of the proposed influence metric for defining the weights of the edges of the graph (Equation 5.1). For this purpose, the influence of each vertex was calculated, and the degree distribution was performed considering the weights of the edges instead of the degree of each node. Finally, an influence distribution graph of the researchers was obtained (Figure 5.4). As the weight of each edge varies from 0 to 1, the influence distribution has also the same variation.

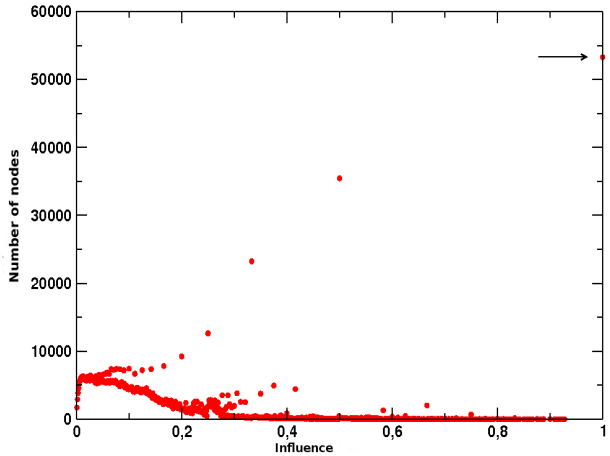


Figure 5.4: Influence distribution of researchers.

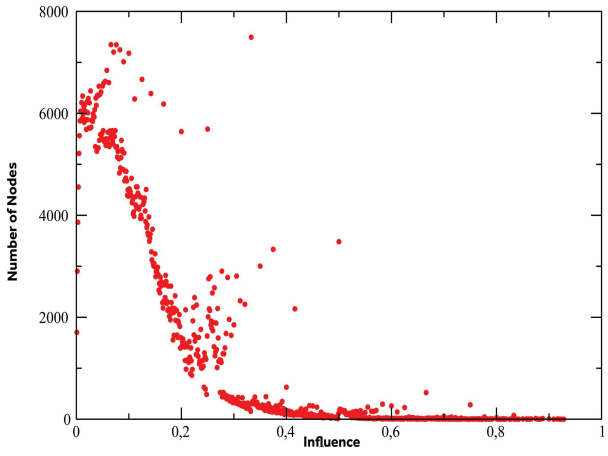


Figure 5.5: Influence distribution in the major connected component.

The degree distribution analysis detected that there are few nodes with high degree. On the other hand, by analyzing the influence distribution in the graph of Figure 5.4, it is possible to see that many nodes have maximum influence (indicated by the arrow). It

is an unexpected behavior based on previous degree distribution analysis, but it occurs because the DBLP network is not fully connected. It is formed by independent connected components that contain few nodes of profound influence. For example, in independent components with only two vertices, both will have maximum influence over each other and still have degree 1. This means that there are several independent groups of researchers that work "isolated", since they do not connect with most of the network.

Although DBLP social network has many independent components of various sizes, there is a dominant component with 1,100,504 vertices. As complex networks focus on analyzing the behavior of networks with a high number of vertices, only the dominant component of the network was considered in this study.

The influence distribution was recalculated to verify the network behavior. The graph in Figure 5.5 shows a result closer to what was expected, that is, a heavy-tail behavior in which few vertices have a high index of influence. The influence distribution is close to a power law, except for some points that appear outside the standard behavior of the graph. These points represent groups of researchers that connect to the largest connected component through a weak link. However, these groups have high connectivity between their members, which results in a set of vertices that does not fit into a power-law behavior.

These results motivated us to use semantic analysis to improve the confidence of the discovered connections and also to discover new connections based on the semantic context of the research. This semantic analysis is presented in Section 5.4.

5.3 COMMUNITY DETECTION IN DBLP

In a next step in the history research, the NetSCAN algorithm was executed in the largest connected component of the DBLP scientific social network. Since the social network is contructed in Neo4j and NetSCAN is implemented as a Neo4j procedure, the execution was done through a cypher query in the database. As a whole, 34,776 research communities were identified, using the parameters: $eps = 1$, $minPts = 5$ and $radius = 1$.

These parameters were defined after some experiments were carried out from part of the database. These experiments allowed observing that in order to identify scientific communities of researchers with direct influence, it is better to adopt $radius = 1$ and a smaller value of *eps*. In this way, it is easy to locate large research communities in which *cores nodes* directly influence the *border points*. On the other hand, to refine these

communities and identify subgroups, it is necessary to increase the value of the radius, since, in general, the researchers in their subgroups are influenced by other researchers.

Considering the large volume of data used in this study, some clusters were selected with the purpose of illustrating the behavior of the proposed algorithm and analyzing the obtained results. Figure 5.6 shows one of the groups defined by NetSCAN, where the centering nodes are represented by blue vertices, and border points by gray vertices. Aiming to improve the visualization of the graph, the influence measures represented in the edges were multiplied by 1000.

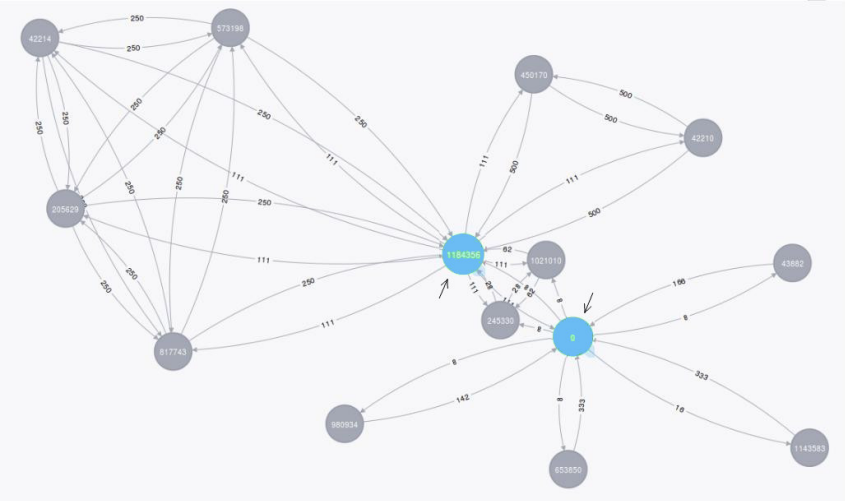


Figure 5.6: A research group identified by NetSCAN.

It can be observed that this group has two influencing vertices and that only two border points are connected in these two vertices simultaneously. This indicates that this group has a great chance of being separated into two subgroups if parameter *eps* is chosen more rigidly, requiring a greater level of influence in the group.

NetSCAN allows for overlaps between research communities, thereby locating multi-disciplinary researchers who are influenced by different research communities. Figure 5.7 shows this situation, where a researcher is associated with two groups.

The blue nodes represent the two communities. The relationships between these vertices with the researchers indicate that they participate in the community. It can be observed that the blue one (indicated by arrow) has connections with members of the two groups (red rectangle nodes), but is sufficiently influenced only by members of the group to the right of the figure and, therefore, participates only in this community. Other vertices of this set also have peculiar properties, for example, the leftmost vertex (338001) exerting such a high influence to the point of attracting a large number of collaborators.

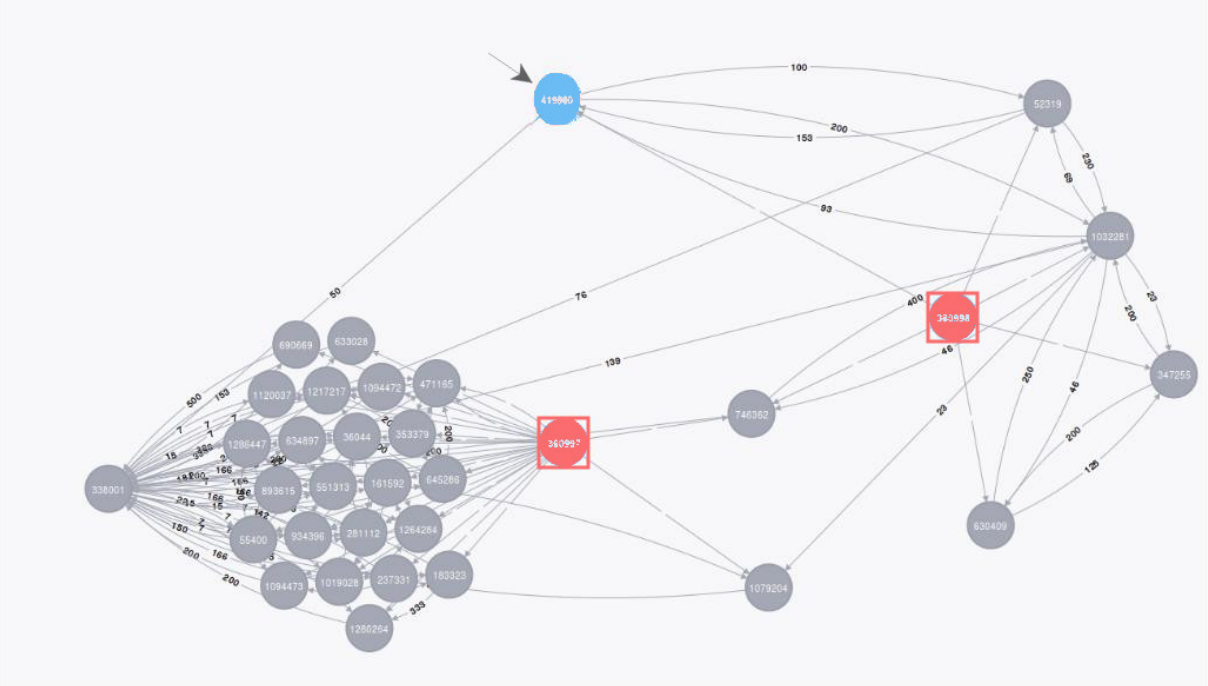


Figure 5.7: Two overlapping communities found by NetSCAN.

Besides this, the central vertex with label 746362, although connected to a few nodes, has a sufficient influence relation to participate in both groups.

Although the results are promising, with this topological analysis it is not possible to state that these communities are real, i.e., if they are semantically compatible. In addition, it is also important to verify if there are other semantic communities that were not identified by the topological analysis.

5.4 SEMANTIC ANALYSIS IN DBLP

In order to analyze if the groups found by the topological analysis in DBLP have characteristics of real research communities, the semantic analysis proposed in Chapter 4 was used.

A group of the network was extracted. In this group, only one of the researchers is considered a core point. In addition, this group does not overlap with other groups.

To start the semantic analysis, the 5 steps proposed in Chapter 4 were performed. The first one is the preprocessing step. Consequently, the first step was to choose a domain taxonomy that would provide terms related to the network domain. Again, the CCS (https://dl.acm.org/ccs/ccs.cfm) taxonomy was chosen, as it provides terms related to the CS area in a hierarchical way. The terms extracted from the CCS were inserted in

NetO ontology as individuals of the Subject class.

In the second step, the keywords in the researchers' publications were extracted and inserted into NetO ontology as individuals of the Keyword class. Since the keywords and concepts of the CCS taxonomy are often spelled differently, it is necessary to map the keywords used by the authors according the CCS taxonomy concepts. NetO ontology is able to perform this mapping through one of its ontological rules (rule 1). We present below two specific examples extracted from this group to illustrate this mapping.

Keyword (Data Mining) $\rightarrow$ *Subject (Data Mining)*

Keyword (technology-education) $\rightarrow$ *Subject (Information technology education)*

After performing the mappings, the researchers were extracted from the network and inserted into the ontology as individuals of the Node class. In addition, each researcher was associated with their respective keywords.

Apart from identifying the semantic context of each researcher, one of the objectives of this analysis was to identify the semantic context of the community as a whole. Thus, to identify topics of interest of the community, an individual community was created, belonging to the **Node** class. The keywords of the individual community were taken from publications that involved more than one member of this group. Finally, after populating the ontology with individuals of the **Node**, **Keyword** and **Subject** classes and performing the necessary *Keyword* $\rightarrow$ *Subject* mappings, the ontological rules (rules 1 to 7 - Chapter 4) were processed by the inference machine and topics of interest of each researcher were extracted. With the inference machine, it was also possible to discover implicit semantic relations, and with these relations, it was possible to discover new connections between researchers (Table 5.1).

Figure 5.8 shows a "word cloud" built from all keywords found. By this "word cloud", it is possible to know the main terms used by the researchers, which evinces this community's interest areas. With emphasis on terms such as "motivation", "comprehension" and "education", there may be indications that this community is possibly interested in the areas of education and learning.

Table 5.1 shows the topics of interest inferred by the ontology for each researcher and for the community as a whole, besides showing the researchers that have similar semantic contexts. The third column of Table 5.1 shows the results of ontological rule 7, which extracts the implicit relations between the researchers of this community based on their

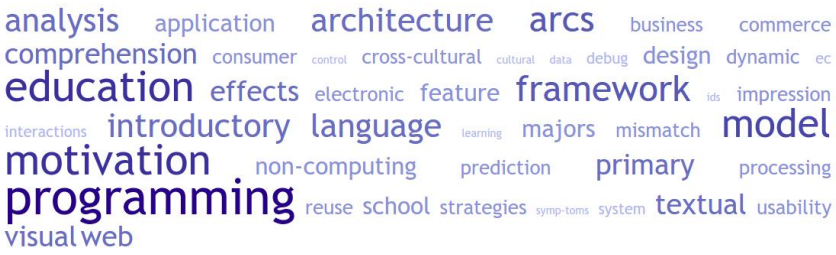


Figure 5.8: Frequent terms found in abstracts of this community's researchers' publications.

Researcher	Topics of interest	Similar context (inferred based on the ontological rules)
Community	Education;computing education;language features;context_specific_languages;user characteristics;Web Interfaces;	R1,R2,R3,R4,R5,R6,R7
R1 (core)	Education;computing education;language features;context_specific_languages;user characteristics;Web Interfaces;	R2,R3,R4,R5,R6,R7
R2	Education;computing education;context_specific_languages	R1,R3,R4
R3	Education;computing education;context_specific_languages	R1,R2,R4
R4	Education;language features;DataMining	R1,R2,R3,R5
R5	Language features	R1,R4
R6	user characteristics;Web Interfaces	R1,R7
R7	Web Interfaces	R1,R6
R8	-	-

Table 5.1: Semantic context of this community and of each researcher.

semantic context.

Also, the data in Table 5.1 show that this community has great interest in the areas of education and education focused on computing. Researcher R1 is the only core point of this community and participates in almost all areas of interest of the group. Researcher R4 is the only one who has interests in Data Mining. Therefore, as this group did not work together on this topic, Data Mining was not considered a topic of interest for the community as a whole. Researcher R8 has few publications and in none of them keywords were specified.

As a result, it can be seen that this community has characteristics of a real research community, acting mainly in the area of education focused on computing and technology.

It is also noticed that R1 has characteristics of an influential researcher in this group, since, except for R8, all other researchers have a similar semantic context with R1. We can also observe a greater similarity of semantic contexts in the sets of researchers R6, R7 and R2, R3, R4.

In this way, through this semantic analysis, with the use of NetO ontology, it was possible to extract the scientific contexts of the researchers and the community involved in order to analyze the area of action of the group and to infer implicit relations between

the researchers of a scientific community.

5.5 OVERLAPPING COMMUNITIES USING TOPOLOGICAL AND SEMANTIC ANALYSIS

Concerning the overlapping communities, quantitative and qualitative analyses of the results of the clustering algorithm were carried out.

5.5.1 OVERLAPPING COMMUNITIES CHARACTERIZATION (TOPOLOGICAL ANALYSIS)

Firstly, a quantitative analysis was performed, and different types of overlapping were observed, referred to as Border Point Overlapping Communities (BpOC) and Core Point Overlapping Communities (CpOC). To characterize the types of overlapping, we considered the existence of two communities C_i and C_j; c_i and c_j as being core vertices of communities C_i and C_j respectively; and $C_i \neq C_j$. The overlapping type BpOC is the case in which these two communities overlap through a border point vertex. Figure 5.9 represents this overlapping type where communities number 303 and 14 are overlapped by researcher 449359.

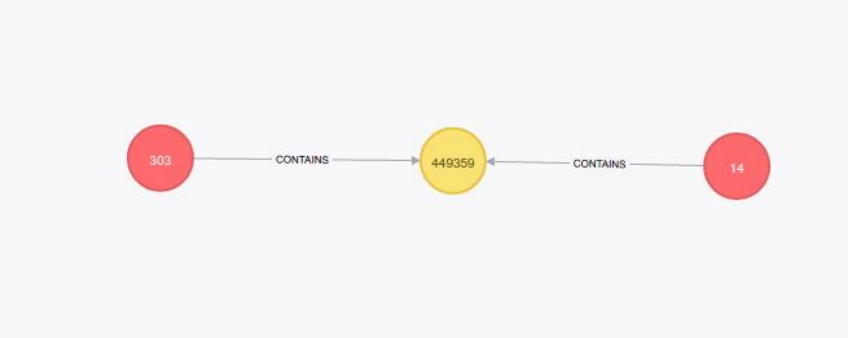


Figure 5.9: Two communities (red) overlapped by a border point (yellow).

The main characteristic of BpOC overlapping is that the overlapped vertex is a border point vertex (BpV) and therefore its inclusion in overlap $C_i \cap C_j$ does not expand and does not change the configuration of $C_i \cap C_j$. BpOC overlapping can also occur when multiple border points are connected to *cores* c_i and c_j. This type of overlapping is defined by $C_i \cap C_j = BpV^n$, where BpV^n is the set of n border point vertices that are influenced by two or more *cores* of two or more distinct communities simultaneously.

In order for such overlapping to occur, a border point vertex (BpV) must be influenced by cores belonging to distinct communities, for example, by *cores* c_i e c_j simultaneously. In this vein, we can say that BpOC overlapping occurs when the conditions defined in (5.3) are satisfied.

$$\begin{aligned} c_i &\in C_i \\ c_j &\in C_j \\ c_i &\notin C_j \\ c_j &\notin C_i \\ IP(e_{BpV}, c_i) &\geq eps \\ IP(e_{BpV}, c_j) &\geq eps \end{aligned} \tag{5.3}$$

The other type of overlapping is CpOC, which occurs when two communities overlap through a *core point* vertex (CpV), as shown in Figure 5.10. In this figure, the overlapped *core* is represented by the blue vertex (indicated by arrow), the clusters are the red rectangular vertices, and the gray vertices are border points influenced by this *core*.

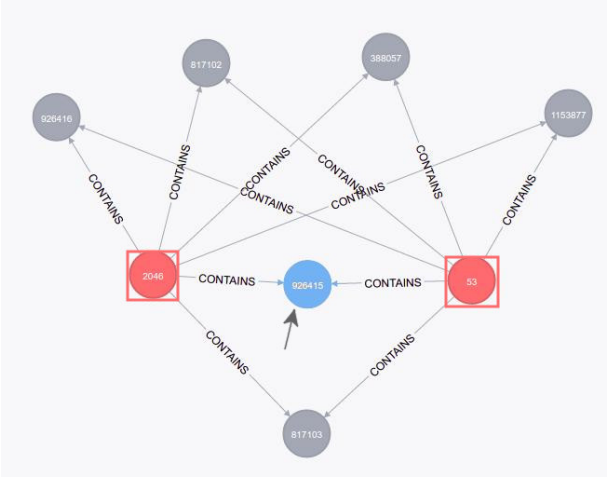


Figure 5.10: Two communities (red nodes) overlapped by a *core* (blue node) and its neighbors (gray nodes).

In CpOC, when a vertex is included in overlap $C_i \cap C_j$, this vertex expands and changes the configuration of $C_i \cap C_j$. This is because the inclusion of a vertex characterized as core point (CpV_l) in $C_i \cap C_j$ implies that the elements influenced by CpV_i are also included in both communities, as shown in Figure 5.10. CpOC can be defined as $C_i \cap C_j = CpV_l \cup CpV_l^n$, where CpV_l is the overlapped *core point* vertex and CpV_l^n is the set of n

neighboring vectors influenced by CpV_l.

Figure 5.11 illustrates the behavior of NetSCAN in detecting CpOC overlapping types. This figure displays six steps of the algorithm for overlapping detection and the expansion caused by the overlapped *core*. To facilitate visualization, only the edges with influence greater than *eps* are represented. In this example, $minPts = 3$ is considered. Figure 5.11 illustrates the following steps:

1. Vertices and network connections to be evaluated by NetSCAN.

2. Vertex 1 (v_1) is chosen randomly. Since v_1 meets the properties of the algorithm ($minPts$ and eps), it is defined as *core*, and community C_1 is created with vertices 1, 2, 3 and 4.

3. Since vertex 4 (v_4) is also core point, its expansion occurs and vertices 5 and 6 are included in C_1. At this point, NetSCAN ends the expansion of C_1 because there are no other core to expand.

4. Vertex 7 (v_7) is chosen randomly. Since v_7 is core, we create community C_2 with vertices 4, 8 and 9. In addition, since v_4 is a core that belongs to another community (C_1), the overlapping type CpOC is characterized between communities C_1 and C_2.

5. Since v_4 is core, its expansion occurs. Vertices 5 and 6 are included in C_2 , also causing the expansion of overlapping $C_1 \cap C_2$.

6. End of the clustering process because there is no other core to be evaluated.

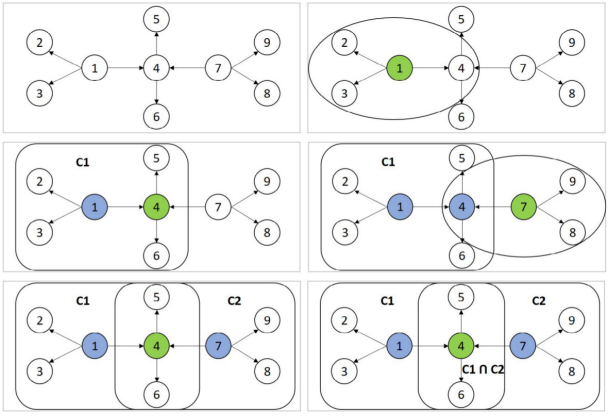


Figure 5.11: Overlapping communities Type CpOC step by step.

Based on Figure 5.11, we can highlight the difference between Border Point Overlapping (BpOC) and Core Point Overlapping (CpOC), considering the inclusion of vertices 5 and 6 in the intersection of communities. If vertex 4 were a border point, this overlapping would be of type BpOC and vertices 5 and 6 would not compose this overlapping.

In addition to the case illustrated in Figure 5.11, other vertices may be of the *core point* type ($CpV_{1..h}$) to be influenced by CpV_l , where $CpV_{1..h} = CpV_1, CpV_2, ..., CpV_h$ and $CpV_{1..h} \subset CpV_l^n$, where h is the number of *core points* influenced directly or indirectly by CpV_l. This causes the inclusion of $CpV_{1..h}$ in $C_i \cap C_j$, the expansion of these vertices and the inclusion of the neighbors influenced by them. This process repeats until all core points belonging to $C_i \cap C_j$ are expanded. Thus, we can extend the overlapping definition of type CpOC to $C_i \cap C_j = \{CpV_l\} \cup CpV_l^n \cup \{CpV_1^n \cup ... \cup CpV_h^n\}$, where $CpV_{1..h}^n$ are sets of vertices influenced by $CpV_{1..h}$.

In the clustering performed in the DBLP network, the two overlapping types previously defined were detected and analyzed. For each type, configurations were found. There are cases involving more than two communities, multiple overlapping border points, and even multiple overlapping cores. Figure 5.12 shows some of these different overlapping types.

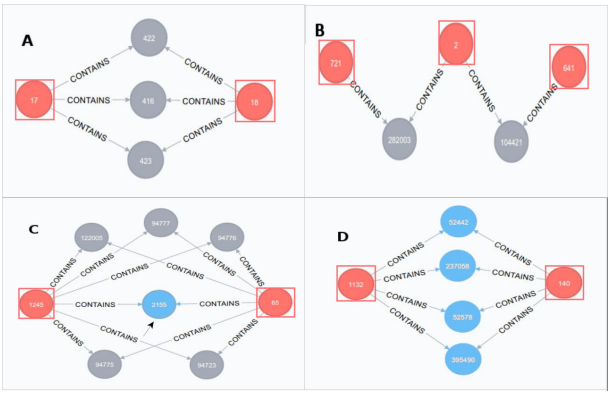


Figure 5.12: Different configurations of overlaps between communities (red nodes) and their participants (yellow nodes).

The configurations shown in Figure 5.12 are (a) three overlapping *border points* in two communities, (b) two overlapping border points in three communities, (c) one *core* (indicated by arrow) and its five neighboring overlapping *border points* in two communities, and (d) four overlapping *cores* in two communities. In Figure 5.12d, only the overlapping *core points* are represented as an easier way of visualizing information.

After finding the overlapping types and the existing configurations, quantitative data were collected in result of the grouping. The obtained results demonstrated that the

number of overlapping communities represents approximately 31.5% of the total communities found. This indicates that interaction between different scientific communities is common. It was also noticed that most of the overlapped nodes are *border points*. This happens because every time a *core* belongs to two distinct communities its neighbors are also associated with such communities.

Based on the analyses performed and with the aid of Figure 5.12, we can affirm that there are elements that represent research-information brokers and others that are central connectors. From these analyses, we can answer the first and second secondary research questions, as discussed in first section of this chapter.

5.5.2 OVERLAPPING COMMUNITIES OVER TIME (SEMANTIC ANALYSIS)

In this section, in order to evaluate the real reason for the occurrence of overlaps, a detailed analysis of the overlapping communities was carried out. For this purpose, a temporal analysis of these communities was performed, aimed at observing their evolution over time, identifying the characteristics of the overlapping.

Through the semantic analysis of the publication history of several overlapping nodes (researchers), three real overlapping motivations were found, namely, change of area of activity, change of research group, and simultaneous action in multiple research groups.

5.5.2.1 Change of area of activity

The first cause of overlapping was the change of area of activity. The analyzed researcher was identified as a border point, and participates in a BpOC overlapping between two communities.

By analyzing the evolution of this researcher over the years, it was found that he/she has publications co-authored with members of one of the communities until the year of 2001. As of 2011, the publications were co-authored only with the members of the other community. In the interval between 2001 and 2011, few publications of this researcher were found. Most of these publications were single-authored papers. Table 5.2 shows a summary of the history of publications with these research groups.

After analyzing the years of collaboration between the researcher and these two com-

	First publication	Last publication with C1	First publication with C2	Last publication with C2
Year	1984	2001	2011	2015

Table 5.2: Temporal publication analysis of this researcher with communities C1 and C2.

munities, a survey of their areas of action was conducted. To do so, the keywords in the publications were collected in the periods 1989–2001 and 2011–2015. The keywords were used to populate the NetO ontology described in Chapter 4. Then, the semantic analysis was conducted to extract the topics of interest of this researcher. The CCS taxonomy was used again to provide the computating related terms.

Table 5.3 shows the topics identified by the ontology in the two periods. In Table 5.3, a change can be observed in the topics of interest of this researcher between the two periods. Only the topic "Human centered computing" was present in the publications in both periods. In addition, the topics of the first period are associated with a large area of Software and its engineering while the topics of the second period are associated with the areas of Artificial Intelligence, Information systems and Security and Privacy.

Topics of interest (1989–2001)	Topics of interest (2011–2015)
Software System Models Software System Structures Human centered computing	Web Mining Natural Language Processing Security Services Human centered computing

Table 5.3: Terms used in the publications in two periods.

In this way, it was considered that the overlapping in this vertex occurred due to the change of area of activity. This overlapping occurred due to the evolutionary analysis of the network. Thus, although it is a real overlap, there is no indication that this researcher is a research-information broker, since the withdrawal of this researcher from the network does not affect the community in which he/she stopped participating.

5.5.2.2 Research group exchanges in the same area of activity

Another reason for this overlap was the change of research group without changing the area of activity. In this case, the selected researcher was identified as core and participates in a CpOC.

Again, to understand the real reason for the overlap, a semantic analysis was conducted involving all the publications of this core. It was verified that the researcher has

publications with members of one of the communities until the year of 2003. From 2004, his/her publications were coauthored only with the members of the other community. Table 5.4 shows more details about the history of publications with these groups.

	First publication	Last publication with C1	First publication with C2	Last publication with C2
Year	1984	2003	2004	2017

Table 5.4: Temporal publication analysis of this researcher with communities C1 and C2.

It can be seen, from Table 5.4, that the analyzed researcher migrated from community C1 to community C2 in 2004. After analyzing the years of collaboration between this researcher and his/her two communities, a survey was carried out on his/her areas of activity. For this survey, we mined digital libraries (mainly IEEExplore) to get the abstracts of his/her publications in the periods 1984–2003 and 2004–2017. The abstracts were again submitted to a semantic analysis and the topics of interest were inferred by NetO ontology. Table 5.5 shows the topics in this researcher's publications during the two periods.

Topics of interest (1984–2003)	Topics of interest (2004–2017)
Signal processing systems;	Signal processing systems; Randomness, geometry and discrete structures

Table 5.5: Terms used on this core researcher's publications in two periods.

As there is a similarity between the topics presented in Table 5.5, it is not possible to conclude that this researcher changed his area of activity. It was considered, therefore, that this researcher made a change of research group without changing his/her area of activity.

Unlike the previous case, this overlapping characterizes a research-information broker. In this case, as the researcher stopped collaborating with one of the communities, there are indications that there was a lack in collaboration between communities.

5.5.2.3 Interacting simultaneously in multiple research groups

The third real reason for the overlap found was the simultaneous activity in multiple research groups. This type of overlapping occurs both with border point vertices and with core point vertices. This overlapping indicates that the researcher influences (if a *core point*) or is influenced (if a *border point*) by both the research communities. To analyze

Topics of Interest C1	Topics of Interest C2
Randomness geometry and discrete_structures	Randomness geometry and discrete_structures; Storage class memory

Table 5.6: Terms used in this core researcher's publications in two periods.

overlapping with these characteristics, we investigate the history of another researcher represented by a *border point* vertex. This researcher participates in a BpOC overlapping type between two communities.

Firstly, a temporal analysis of this researcher's publications was performed with each community. It can be seen in the graph of Figure 5.13 that there is no interruption in the activity of this researcher with his/her two communities. It is worth noting that between the years 2012 and 2016 this researcher collaborated with the two communities simultaneously.

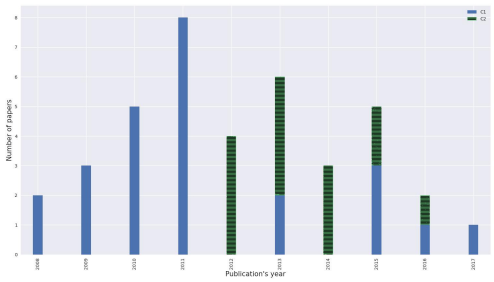


Figure 5.13: Temporal analysis of publications with two communities.

In addition, no publications were found among other researchers in C1 and C2. In this way, we can consider that the two communities are indeed distinct and overlap through the analyzed border point.

After characterizing the simultaneous activity in the two research groups, an analysis of the researcher's area of activity was carried out. The purpose of this analysis was to verify if the researcher works with the same themes in both communities. For this, the abstracts of the publications used in the previous analysis were collected again from digital libraries and submitted to a semantic analysis. This analysis applied the same criteria as those used in the previous cases.

Table 5.13 shows the topics of interest inferred by NetO ontology for this researcher's publications in each community.

Again, the context of the publications between the two communities are similar. Therefore, there are indications that this researcher works with the same research themes in both

communities. In addition, the researcher can be characterized as a research-information broker because, to a certain extent, he/she is responsible for maintaining the exchange of information between the two scientific communities in which he/she participates.

5.6 DISCUSSION

As presented in this **History Research**, through topological and semantic analyses, scientific communities were identified, in addition to influential researchers, researchers participating in more than one community, and new relations among researchers based on their semantic context.

These results can help in the decision making of communities or individual researchers. Discovery of new connections through semantic context similarity can be used to encourage collaboration between researchers who have the same interests. In the case of communities, the identification of influential researchers is important, since any change in their behavior can impact, positively or negatively, the collaboration among the other members of these communities. Another factor is the identification of research-information brokers, who are researchers responsible for establishing collaborations among multiple distinct communities. Identifying these people can help encourage new collaborations between the communities involved. It is also important to mention that research-information brokers no longer have active participation in the network, as this can cause disconnection between communities and hinder future collaborations.

In order to verify if the aims were achieved, two research questions had been previously established, based on the main research questions stated in Chapter 1. These research questions were divided into four secondary questions and each of them are resumed below.

(i) *How to identify researchers who help maintain the network connectivity, disseminating information and linking research groups/subgroups?*

To evaluate this question, three secondary questions had been proposed, each one evaluating a certain aspect.

RQ1: *Are there researchers who work in two or more communities simultaneously, characterizing research-information brokers?*

In Figures 5.12a and 5.12b, we can observe researchers who participate in two communities simultaneously, and the temporal and semantic analyses of these overlapped researchers enabled us to identify the real causes of these overlaps.

The overlapped researchers who moved from one community to another can be considered research-information brokers because of their potential to establish communication between their communities. There is also the case of overlapped researchers who are already acting simultaneously in more than one community. This one shows a very positive characteristic of a research-information broker, promoting the collaboration between two different communities.

Although these researchers are not central connectors, because they do not influence a group of researchers, they influence and are influenced by more than one community. In this case, communication and collaboration between communities takes place through them.

RQ2: *Are there researchers who are largely connected to another researcher, characterizing central connectors?*

Through NetSCAN, it was possible to identify centralizing elements that are widely connected to other researchers, as illustrated in Figure 5.6. Given the high degree of influence that these researchers exert on other researchers, they are considered central connectors. These central connectors are represented by the *core points* in the network and are identified automatically by NetSCAN. The degree of influence is related to the *eps* parameter on NetSCAN and can be set to define how largely influential the central connectors are.

Generally, central connectors influence their own community, but in some cases they can influence more than one community simultaneously. So, an analysis was conducted to identify overlaps in the network over time, as shown in Figure 5.12a and 5.12d.

The semantic analysis allowed observing that because central connectors are largely connected to other researchers, they can define the semantic context of their communities. This characteristic can be observed in Table 5.1.

RQ3: *Does the use of cluster analysis help discover real scientific communities considering the activities developed by researchers?*

The topology and semantic analysis showed that the researchers belonging to the same community not only have influence on each other but also have a similar semantic context. Table 5.1 shows that by analyzing the semantic context of these researchers it was possible to discover new connections that were implicit, showing that the researchers grouped by NetSCAN had similar interest and consequently were more likely to keep working together in the future.

(ii) *How to discover semantic connections (research interests) between researchers, also including their connections based on their scientific context, even though such connections are not explicit?*

Specifically, to answer this question, we need to answer RQ4: Does the use of semantic analysis reinforce connections and discover implicit relations?

The use of NetO ontology enabled us to identify implicit semantic relations, and scientific contexts of the researchers and their communities. By the analysis of Table 5.1, it can be seen that NetO ontology was capable of finding and matching the researcher's semantic context. NetO ontology reinforces connections and proposes new ones by creating a hasSameInterest semantic relation between researchers with similar contexts.

The semantic analysis can be enriched by adding more sources of data and other types of collaboration between the researchers, such as citations or research projects. As long as the new data is related to the domain taxonomy the NetO ontology is able to integrate and use it to discover new relations.

The historical research and the answers to the research questions indicate that NetSCAN, NetO ontology, and the network analysis were able to identify relevant context information about the scientific network.

6 SOFTWARE DEVELOPMENT SOCIAL NETWORK EVALUATION

In this chapter we conduct another evaluation for the NetSCAN algorithm and NetO ontology. The main difference from the previous chapter is that in this evaluation we will perform the semantic analysis before applying the NetSCAN algorithm. In addition, we want to evaluate the use of **SpecificTopic** class in NetO ontology, which have not been used in previous evaluation.

For this, a different real social network is used in the context of software development. The first objective of this study is to find communities of developers with strong connections and similar topics of interest. The second goal is to understand the meaning of the *core* nodes in this new context. Based on the previous results, by using semantic analyses followed by NetSCAN execution we expect to find collaborative groups, expert developers and their topics of expertise. Again, the scope of this evaluation was based on the GQM method, described as follows: *"**To analyze** semantically and topologically the developer social network and its overlapping communities **for the purpose** of providing information for decision-making concerning information brokers and central connectors **in relation to** software development communities and expert developers **under the point of view** of software development decision makers **in the context** of collaborative environments of developers"*. Therefore, two Research Questions were stated, also based on the research questions previously raised on Chapter 1:

- (i) How to discover semantic communities of developers?
  - RQ1: Does the use of NetSCAN and NetO allows the detection of communities with high internal connectivity?
  - RQ2: Are the members of detected communities semantically aligned with their assigned topics of interests?
- (ii) How to identify expert developers and their topics of expertise?
  - RQ3: Are there developers who work in two or more communities simultaneously, characterizing information brokers?

– RQ4: Are the *core* nodes developers with relevant contributions in the network, characterizing central connectors?

The first research question is important to assess the cohesion of detected communities and to explain whether the members are in fact involved with their communities' topics of interest. The second investigates the meaning of *core* nodes in the network by questioning if they are important people with characteristics of being information brokers or central connectors.

Similarly to the previous chapter, we chose to use the History Research for the evaluation method since we do not have control over the events in the Q&A forum. As sources of evidence archival records were used, extracted through the StackOverflow Api, and context-aware data collected from the StackOverflow website.

We first describe the collaborative model used for building the social network, and then we explain the use of NetO and NetSCAN in this model.

6.1 STACKOVERFLOW COLLABORATIVE NETWORK MODEL

One way to study interactions between software developers is through analyses of question-answer forums (Q&A) where many users are actively collaborating and helping each other. Yahoo! answers, Quora and StackOverflow are some examples of Q&A forums that generates huge amounts of high quality and highly reusable information (MENG et al., 2016). Among them, StackOverflow is the most popular in software development context by having a large number of users and a high answer rate (MAMYKINA et al., 2011). Through the question-answers mechanism, Stackoverflow website promotes collaboration and knowledge exchange between it's users. Some important characteristics of this forum are: the use of tags to define the domain of each conversation; and the scores of both questions and answers, which helps determining the relevance of each contribution.

The activity starts when one user creates a post to ask a question in the forum and any registered user is allowed to answer the question. Each user can give multiple answers to a same question and each of these answers contains up to 5 tags. The answers also have a score, which is assigned by other users in the forum.

In general, Q&A forums do not imposes explicit relationships between users and thus any user is allowed to answer questions asked by any other use, even though they are not

explicitly connected. For this reason, to construct a social network based on Q&A data, the relationships have to be inferred in some way based on the users' interactions. There are three main possibilities to define the graph topology in this scenario (RÍOS; MUÑOZ, 2014):

- All-previous-reply: every post in a thread is a response to all previous posts in this thread. Accordingly each node connects to any node representing the previous replier of the common interest thread.
- Last-reply-oriented: every post in a thread is a response to the last post on this thread. Thus each node connects to only the node which was the last replier in the common interest thread.
- Created-oriented: every post in a thread is a response to who created the thread. Accordingly each node connects only to who created its interesting thread.

The *All-previous-reply* representation generates a very dense graph since all possible connections are considered and this can be inconvenient for community detection methods. In *Last-reply-oriented* the most important link may be unrepresented, which is the link connecting the question creator to the repliers. Thus, for our model we chose to use the *Created-oriented* representation, whereas it has the minimum density among other representations and contains the most important link in the network (KIANIAN et al., 2017).

Following the *Created-oriented* representation, edges in our model were inferred from answers extracted from StackOverflow. In order to identify the topics of interests of communities and developers, we chose to use different relationship types for each tag in StackOverflow. Lastly, to consider the relevance of each contribution between the users, the edge weights are calculated using the *score* obtained by the answers.

In this vein, to study developers interactions in this context we have formally modeled a social network where the nodes represent users and the edges represent all answers given by the source node to the target node (HORTA et al., 2018a). The network is represented by a bidirected graph $G = (V, E)$, where $V = \{v_0, v_1, ..., v_{n-1}\}$ is the set of vertex (nodes), and E represents the set of edges $e_{ijt} = (v_i, v_j, t)$ which connects users v_i e v_j in a tag t. As a user can give many answers related to the same tag, $R_{ijt} = \{r^0_{ijt}, r^1_{ijt}, ..., r^{l-1}_{ijt}\}$ is a

set of l answers related to tag t between users v_i and v_j such that r^k, for $0 \leq k < l$, has an integer score.

In order to measure collaboration between users in each tag, the edge weights represent the impact of all contributions that user v_i gave to v_j in a tag t, defined by $IP(e_{ijt})$.

To calculate $IP(e_{ijt})$, we first sum the scores of answers given by user v_i to v_j. Equation 6.1 shows how this sum is calculated where $S(r^k_{ijt})$ is the $k\acute{t}h$ answer score from v_i to v_j in tag t and l is the total number of answers given by v_i to v_j in tag t.

$$Sum_{ijt} = \sum_{k=0}^{l-1} S(r^k_{ijt}) \tag{6.1}$$

Equation 6.1 is normalized by dividing the Sum_{ijt} by the total contributions received by v_j in tag t. Equation 6.2 defines the edge weight $IP(e_{i,j,t})$ where $\parallel N(j,t) \parallel$ is the total score received by v_j in tag t.

$$IP(e_{ijt}) = \frac{Sum_{ijt}}{\parallel N(j,t) \parallel} \tag{6.2}$$

Because of $Sum_{ijt} \leq \parallel N(j,t) \parallel$ we have that $0 \leq IP(e_{ijt}) \leq 1$. This way, when $IP(e_{ijt})$ approximates to 1 the contributions from v_i to v_j in tag t are considered to be more relevant. On the other hand, when $IP(e_{ijt})$ approximates to 0, these contributions are considered less relevant. If $IP(e_{ijt}) = 1$ then it means that v_i is the only user who contributed in a positive way with v_j in this tag. The value of $IP(e_{ijt})$ can also be 0 in case that v_i has contributed with v_j but the sum of these contributions is zero.

To summarize, the social network model is represented by a heterogenous graph with directed and weighted edges. Figure 6.1 shows an abstraction of this model, where a user u_i has contributed with user u_j with answers in different tags. The size of the arrows indicates the edge weights, which represents the relevance of contributions in each tag.

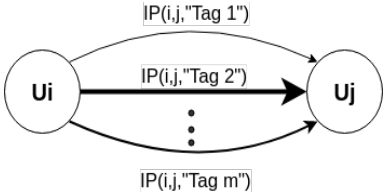


Figure 6.1: Social network model abstraction with various types of directed and weighted edges. Weighted are indicated by arrow size.

Based on this model we have extracted a dataset from the StackOverflow Q&A forum

that contains 565680 users and 618726 questions (posts). The questions are social objects that allow user interactions.

This dataset was first preprocessed with NetO ontology. After the preprocessing the enriched dataset was used to contruct the network in Neo4j.

6.2 SEMANTIC ENRICHMENT IN STACKOVERFLOW NETWORK

In order to semantically enrich the StackOverflow network, a semantic analysis was performed based on the 5 steps described in Chapter 4. In this scenario, the semantic information is contained in questions and answers, which are the tagged social objects that represents collaboration between developers in the Q&A forum. These social objects and their tags are represented in NetO by **Node** and **Keyword** classes respectively. Table 6.1 describes what each class represents and how they are related.

Table 6.1: NetO classes in software development context

Class	Represents	Instatiated by	Related to
Node	Social objects	Automatic extraction process	Keyword through "hasKeyword"
Keyword	Tags of social objects	Automatic extraction process	Subject through "hasTopic"
Subject	Relevant concepts	Predefined in the ontology	Topic through "hasTopic"
Topic	Topic of interest	Predefined in the ontology	SpecificTopic through "hasSpecificTopic"
Specific Topic	Specific topic of interest	Predefined in the ontology	-

As defined in Chapter 4, first step of semantic analysis is the identification of a knowledge domain base. As we do not have a consistent taxonomy for this context, such as the CCS used in Chapter 5, we have used definitions provided by StackOverflow. Because StackOverflow has a tagging system we have chose to extract the individuals *Subject* from the tags used in questions and answers. In this way, some examples of *subjects* and *topics* are *"python"*, *"mysql"* and *"android"*.

Through StackOverflow website, it was possible also to find some relations between tags that can be used for defining *specific topics*. For example, looking at the website we found that *"scikit-learn is a free and open-source machine learning library written in Python."* and by that we inferred that posts with *scikit-learn* tag are related to a *python-for-machine-learning* **SpecificTopic**. Many similar relations could be found in the "tag

info" section of the website and some examples are: programming language frameworks, libraries, IDEs and synonyms.

After defining the NetO domain knowledge, we input the social objects (questions) in the ontology for the enrichment process to find topics and specific topics. This was achieved through an automatic process using java application and Jena library. This application was responsible to instantiate the social objects in the ontology and to run the inference machine for each of them. To illustrate the topic extraction process, we collected a post from the StackOverflow website. The post represents a social object with the set of tags {"**pandas**", "**machine-learning**"}, as shown in Figure 6.2.

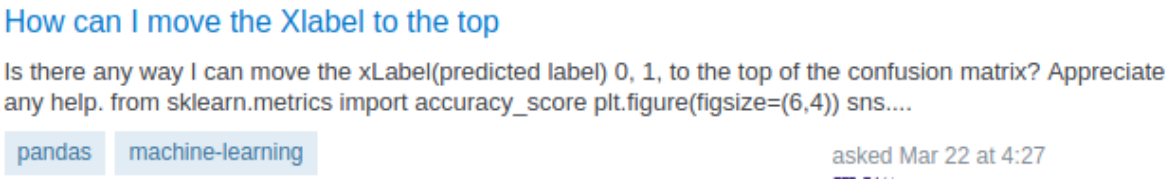


Figure 6.2: Post extracted from StackOverflow forum

Suppose that, in this example, we are not interested in finding specific libraries as topics but instead we want to know programming languages and their usage purpose. In this case, the "**pandas**" tag is not a topic of interest but can be used for the inference of a "**python**" subject. Also, suppose that the "**machine-learning**" tag is relevant as a topic of interest and that "**python-for- machine-learning**" is a relevant specific topic for the context. Table 6.2 shows the individuals of this example in description logic.

Based on this example, we can expect that, when the NetO ontology receives this social object as an input, it should output the object assigned with the topics "python" and "machine-learning" and the specific topic "python-for-machine-learning". The result when executing the inference machine with this input is shown in Figure 6.3 and the step-by-step procedure is described below.

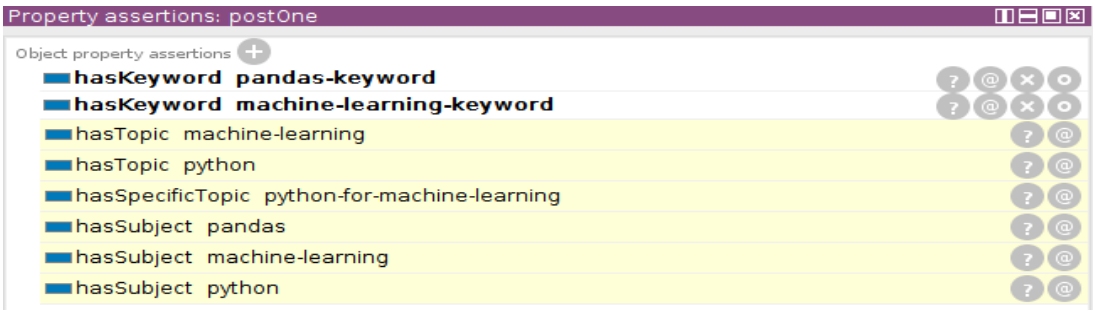


Figure 6.3: Inferred topics for the example individual postOne

First, NetO recognizes that "**postOne**" has the keywords "**pandas**" and "machine-learning". As "pandas" is a *subTopicOf* "**python**", the "**python**" subject is assigned to "**postOne**". As "**python**" and "**machine-learning**" are predefined as topics, the ontology assigns both topics to "**postOne**". As "**python**" and "**machine-learning**" topics forms a specific topic "**python-for-machine-learning**", the specific topic is assigned to "**postOne**".

Table 6.2: Example of individuals in description logic

machine-learning machine-learning : Topic machine-learning : Subject
machine-learning-keyword machine-learning-keyword : Keyword name (machine-learning-keyword "machine-learning"^^ http://www.w3.org/1999/02/22-rdf-syntax-nsPlainLiteral)
pandas pandas : Subject subTopicOf(pandas, python) name (pandas "pandas"^^http://www.w3.org/1999/02/22-rdf-syntax-nsPlainLiteral)
pandas-keyword pandas-keyword : Keyword name (pandas-keyword "pandas"^^http://www.w3.org/1999/02/22-rdf-syntax-nsPlainLiteral)
postOne postOne : Node hasKeyword(postOne, pandas-keyword) hasKeyword(postOne, machine-learning-keyword)
python python : Topic python : Subject hasSpecificTopic(python, python-for-machine-learning) name (python "python"^^http://www.w3.org/1999/02/22-rdf-syntax-nsPlainLiteral)
python-for-machine-learning python-for-machine-learning : SpecificTopic

The output is an enriched dataset containing social objects labeled with topics and specific topics. With this enriched dataset we have created the semantic social graph based on user interactions through the social objects. The graph was represented in the Neo4j database, and the relationship weights were calculated considering the previous defined collaborative model. The result is a weighted and directed graph where relationships have semantic labels related to topics and specific topics discovered in the previous step.

In next section we explain the execution of NetSCAN algorithm in this enriched semantic graph.

6.3 DETECTING COMMUNITIES OF DEVELOPERS WITH NETSCAN

Because our model is represented by a heterogenous graph with multiple relationship types we have defined a strategy in the form of a framework to execute NetSCAN in separated homogeneous subgraphs. This framework has five modules, as shown in Figure 6.4.

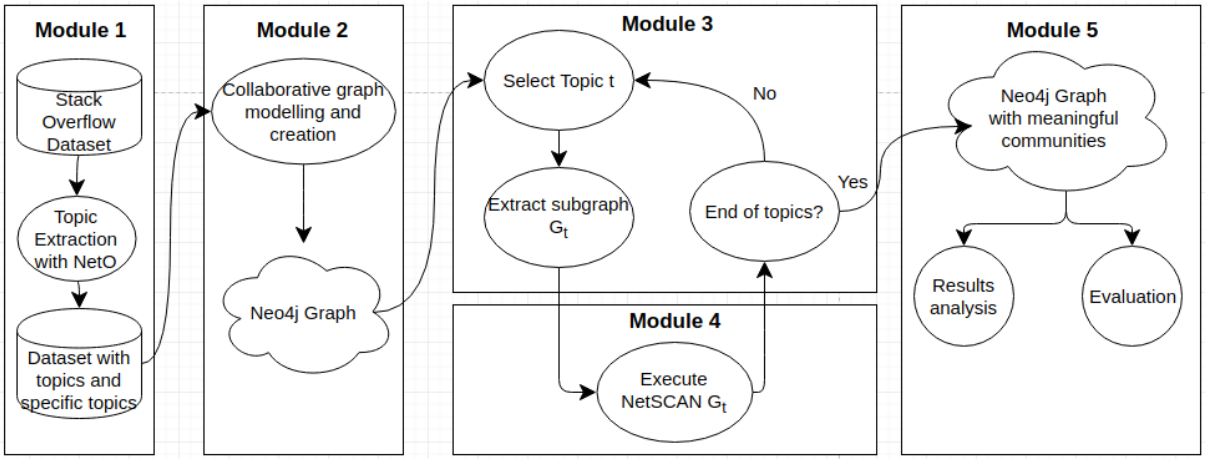


Figure 6.4: Framework to detect semantic communities in the StackOverflow network

The five modules and how we applied each of them in the network stackoverflow are described below.

Module 1. Data preprocessing and enrichment: topics are extracted from social objects and processed to generate an enriched dataset with topical social objects. This step was performed with the semantic enrichment explained in the previous section using NetO ontology.

Module 2. Social network modeling: a social graph is created based on user interactions through the social objects. In our execution the graph was represented in the Neo4j database, nodes and relationships were inferred using the *Created-oriented* representation defined in Section 6.1 and edge weights were calculated considering the defined collaborative model, also in Section 6.1. The result is a weighted and directed graph where the relationships have semantic labels related to the topics and specific topics discovered in the previous module.

Module 3. Social network partitioning: the network is divided into topical subnetworks. Each of these topical networks is related to a single topic. Nodes are allowed to be in multiple topical subnetworks as social network members tend to have multiple interests.

Module 4. Link-based community detection: link-based community detection is ap-

plied to each topical subnetwork and the communities found are labeled with the respective topic. In our case, NetSCAN was applied on each of these topical subnetworks to find topical communities in the software developers' network. The output of this module is the social network and its semantic communities represented in Neo4j.

Module 5. Network analysis: In this final step the results can be analyzed, evaluated and visualized for knowledge discovering. Next section presents the result analysis for the stackoverflow network.

By the end of this framework execution the communities found by NetSCAN are represented as *cluster nodes* in Neo4j and the expert users are *User nodes* marked with the *core* attribute. Because we have enriched the network before executing NetSCAN, the *nodes* carries a label attribute that helps identifying its topic of interest.

6.4 RESULT ANALYSIS

To analyze the results obtained by our approach, we first conducted an exploratory study to investigate the communities' shapes, the existing overlaps and the *core* nodes role. We also analyzed characteristics of the communities with specific topics and how they are related to communities of more general topics.

We first noticed that most of found communities has a star shape, which is a common pattern found in Q&A forums, as previously mentioned by (MENG et al., 2015). Star-shaped communities are characterized by having most of their edges connected to a central node (hub) and peripheral nodes are not interconnected. To illustrate that, Figure 6.5 shows two communities (red circles with arrow) where the first one is a mysql community with a single core node (green rectangle) and the second one is a java community with two core nodes.

It can be seen from Figure 6.5 that most of the relations inside the communities have their origin in the central nodes, which are the core nodes detected by NetSCAN algorithm. This shows that core nodes act as hubs in their star-shaped communities and represent users who give a great number of relevant answers in the forum.

Some communities are related to general topics of interest (ex: "mysql" and "java"), such as those in Figure 6.5, and others are related to more specific topics like "python_for_data_science", as shown in Figure 6.6. The specific topic communities tend to have fewer members and are often subgroups of general topic communities.

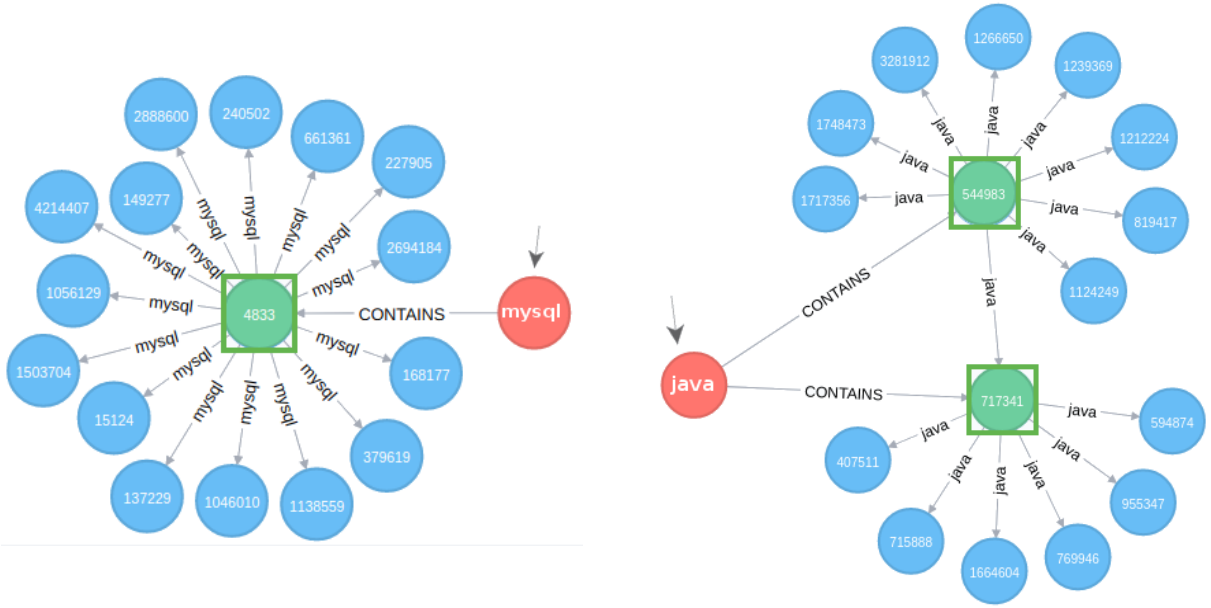


Figure 6.5: Two communities (red with arrow) with general topics and their core nodes (green rectangles)

This happens because NetO ontology combines general topics to find specific ones, so the nodes involved in specific topic relationships are also involved in general ones.

Further interesting findings are related to overlaps in those communities. There are many different types of them, such as overlaps in communities with the same topic, overlaps in different topics, overlaps in general topic and specific topics. As can be seen in Figure 6.7, some of these communities are overlapped by *core* nodes.

The overlaps in different topics involving a core node means that this user gave many relevant answers in two different technologies, which indicates a multidisciplinary characteristic. When the involved communities are from a specific topic, it might also indicate a preference for certain technologies depending on the application purpose. For example, the overlapped node in Figure 6.7.b participates at the same time in a "java_for_web-services" community and a "python_for_data_science" community. This can be used as evidence that this user prefers using java for the purpose of web development but uses python for data science tasks.

The overlaps between two different topics may also indicate that they have a strong relation. For example, the most common overlaps involve topics that are clearly related to each other such as "javascript" and "jquery" or "c" and "c++". However, there are other more interesting cases like "php" with "mysql", which are often used simultaneously, and "javascript" with "android", which can characterize an interest in mobile development with

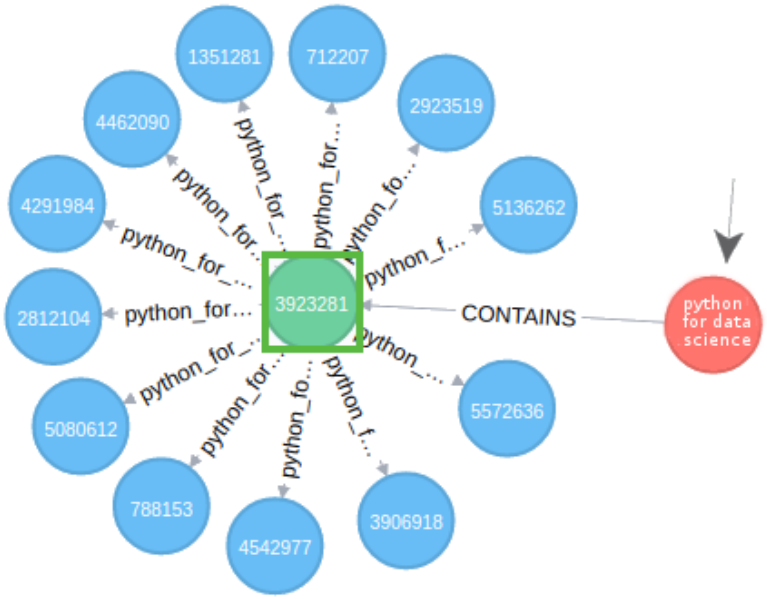


Figure 6.6: A community (red with arrow) with specific topic of interest.

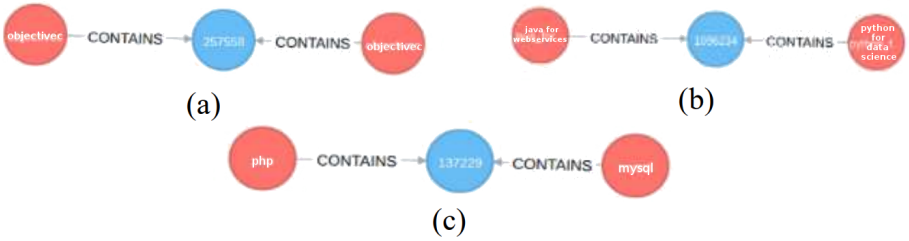


Figure 6.7: Three types of community overlaps: (a) two objective-c communities, (b) a java-for-webservices with python-for-data-science (c) a php with mysql overlap

hybrid technologies. Table 6.3 shows the number of shared users between communities of different tags.

Table 6.3: Developers overlapped in communities of different tags

Tag 1	Tag 2	Shared users
javascript	jquery	9979
html	javascript	4804
css	html	4389
android	java	2476
...	...	...
python	rubyonrails	1
iphone	mysql	2

This exploratory analysis showed that by using the proposed framework we were able to find meaningful communities in software development network. The proposed NetO ontology enabled finding communities with general and specific topics of interests. By using the NetSCAN algorithm, we were also able to detect important users in the soft-

ware network and many overlaps in communities, indicating multidisciplinary users and correlated topics.

In the following sections, we perform quantitative and qualitative analysis aiming to answer our research questions. We first focus in finding whether users are strongly connected with other members in their communities and if they are still active in their communities' topics. Then, we analyze whether the *core* nodes represents expert developers in the network.

6.4.1 EVALUATING THE COMMUNITIES' CONNECTIVITY

To start answering the research questions RQ1 related to the communities' quality, we have first evaluated the cohesion of the found communities. For this, we calculated the silhouette values (TAN et al., 2005) for each community to measure whether their members have better internal connectivity than external connectivity.

The silhouette value for a single node is calculated as follows defined in Equation 6.3, where b_v is the average edge weights of node v to all external nodes, and a_v is the average edge weights of node v to all internal nodes. Because this silhouette value considers the edge weights as being the distance between the nodes, this equation favors clusters with low weights in internal edges. On the other hand, in our model the edge weights represent collaboration between users, so we want clusters with high internal edge weights. Thus, we have to change the numerator in Equation 6.3. Equation 6.4 shows the silhouette value used for our evaluation criteria.

$$s_v = \frac{b_v - a_v}{max(b_v, a_v)} \tag{6.3}$$

$$s_v = \frac{a_v - b_v}{max(b_v, a_v)} \tag{6.4}$$

The silhouette index for a community $S(C_i)$ is the average value of s_v for all of its members, as shown in Equation 6.5. The range of $S(C_i)$ is [-1,1] where high positive values indicate good cluster quality. Negative values indicate that some clusters members have the external connectivity superior than the internal connectivity, which is an undesired

behaviour.

$$S(C_i) = \frac{\sum_{v \in C_i} S_v}{|C_i|} \tag{6.5}$$

A boxplot was used to show the silhouette indexes of the communities found by our approach and is shown in Figure 6.8.

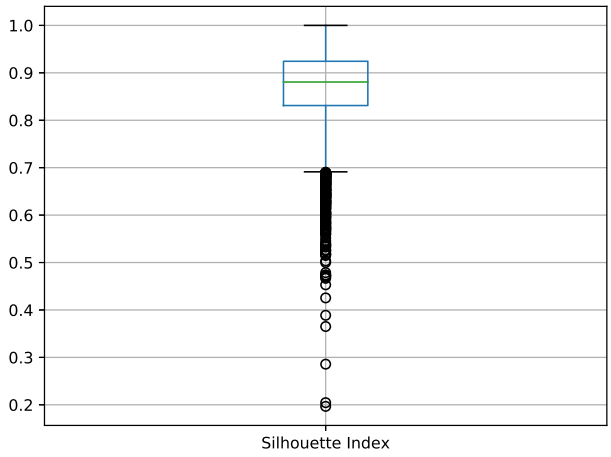


Figure 6.8: Boxplot with silhouette indexes for the communities found

As can be seen in the boxplot, the average value is 0.88, which indicates high quality communities. Some communities have silhouette index equals to 1, which is the highest possible value and means that all positive edges of their members are internal.

It was observed that outliers with low silhouette indexes are communities with multiple overlaps. For example, the lowest outlier represents a community overlapped with 4 other communities. Because the members of these communities are also members of multiple other groups, their external connectivity increases, and their silhouette values decrease.

Although some outliers have a low silhouette index, none of them have a negative value. This shows that all communities have a better internal connectivity than external connectivity and helps answering **RQ1**:

RQ1: *Does the use of NetSCAN and NetO allows the detection of communities with high internal connectivity?*

This quantitative analysis showed that our approach has detected high quality com-

munity in terms of topological structure. On average the silhouette index is 0.88, some communities have the maximum possible value and all of them have positive values.

After checking the topology quality of the communities, in next section we evaluate if communities and their members are compatible in semantic terms.

6.4.2 EVALUATING THE TOPICS OF INTEREST OF COMMUNITY MEMBERS

To investigate whether the users were still active in their communities' topics, the tags within their answers were extracted and analyzed.

According to our analysis in Section 6.4 there are three main types of cores: single community core, overlapped core and specific topic core. For each type, a different set of rules were used to determine whether or not a user was aligned to the related topic.

The first type represents users belonging to a single general topic community. These users belong to a single community and have a single topic of interest. Then, they are considered to be semantically aligned if any of their recent used tags are related with this single topic.

The second type consists of users in a specific topic community. Because specific topics are composed of multiple general topics, these users are considered to be semantically aligned only if they have a set of tags that indicates the specific topic. For example, a user is aligned in the topic "java_for_android" if he/she used a set of tags "java", "android" or "java8", "android- studio" and so on.

Finally, the third type contains users belonging to multiple overlapped communities. As they have multiple communities, they also can be assigned to multiple topics. They are considered to be semantically aligned with their topics if the tags they used are related to all of these topics.

We named these types as A, B and C, respectively. After defining them, we collected the answers from users of each type through the StackOverflow Api. Then we extracted the set of tags from each answer and applied the previous conditions in each type.

The proportion of semantically aligned and active users was the following: 90% for type A, 80% for type B, and 50% for type C.

It was found that 90% of the users in type A continued to respond questions about the related general topic and the exception are users who have no recent answers in the

forum. In type B, 80% of members answered questions with tags that can be combined to indicate specific topic.

Type C has the lowest number of active users. Although most of these users are still active in some of the involved topics, only 50% have recent answers in all of them. This might indicate that some overlaps are caused by users who changed their interests over time and stopped answering questions from an old topic of interest.

This analysis on three different types of core nodes shows that many users are still interested and active in the topics assigned to their communities. Thus, this is an evidence that our approach has found meaningful communities labeled with relevant topics and helps answering **RQ2**:

RQ2: Are the members of detected communities semantically aligned with their assigned topics of interests?

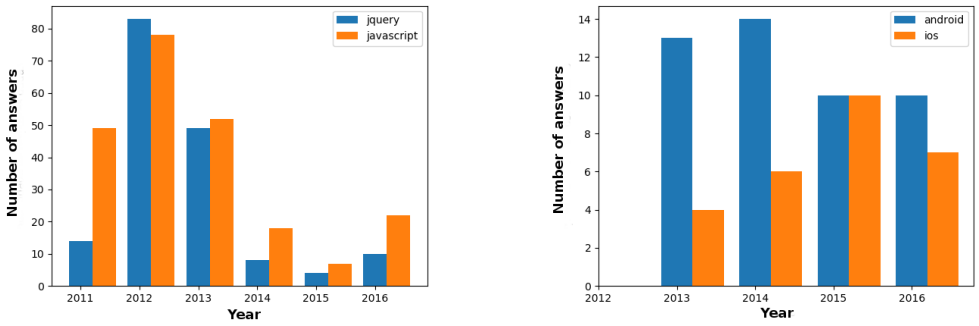
After investigating the recent activity of three different types of important members we have found that most of them are still interested and active in the topics assigned to their communities. However, we argue that a detailed temporal analysis could be useful to explain why the overlapped core nodes have a lower activity rate in some of their topics of interest.

6.4.3 OVERLAPS TEMPORAL ANALYSIS

To investigate what causes a developer to be overlapped in multiple communities we have conducted a temporal analysis over the overlaps. To this, we collected and analyzed the answers given by overlapped users in each of their communities. Two main reasons were found: (i) the overlapped developer is active in both communities and; (ii) the overlapped developer has switched from one community to another.

To show this behaviour we have selected and detailed the history activity of three users. Figure 6.9(a) shows the activity of an user in multiple communities of similar topics (javascript and jquery) and 6.9(b) shows the activity of an user in communities of different topics (android and iOS).

In these figures, both cases shows that the overlapped users have remained active in all their communities. This constant activity characterizes the first reason of overlapping and



(a) Active developer in communities of similar topics (javascript and *jquery*)

(b) Active developer in communities of different topics (android and *iOS*)

Figure 6.9: Developers activity history in multiple communities

can be used to identify highly interested developers 6.9(a) or multidisciplinary developers 6.9(b). The second reason of overlapping is shown in Figure 6.10. In this case the user started contributing in a *"java"* community but after some time he moved to the *"chsarp"* one. Therefore, even though this developer has some competence in "java" he is probably more interest in collaborating with *"csharp"* activities. This shows that some overlaps may indicate an interest change that should be considered in case of recommending the overlapped node to future tasks.

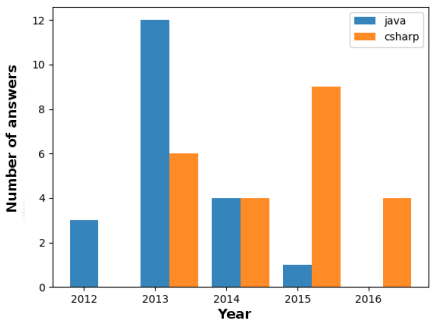


Figure 6.10: Developer change of interest.

There are also cases of expert developers participating in more than two communities, reaching a maximum of 12 communities in a single user. However these are less recurrent cases and thus it is difficult to understand their real causes.

These results allows us to answer RQ3: *Are there developers who work in two or more communities simultaneously, characterizing research-information brokers?*

The temporal analysis shows that by investigating the overlapping communities it is possible to find multidisciplinary developers as well as their interests and competences.

These overlapped nodes can represent active developers in multiple communities, characterizing information brokers. Also, some overlapped users can be in the process of a change in their interests, which was also indicated by the semantic evaluation in the previous section. Locating these people in time can be useful to motivate them to keep contributing in the original community or to help them by facilitating the transition. In next section we conduct an analysis to check whether the *core* nodes detect by our proposal have in fact a good performance in the question-answer context and can be considered as expert developers.

6.4.4 ANALYZING CORE DEVELOPERS

In this section we evaluate whether the *core* nodes detected by our approach are active and skilled developers in their topics of interest. Our hypothesis is that the *core* nodes represents expert developers in the network. This evaluation is based on recent data collected through the StackOverflow API. The data contains recent answers from users and it can be considered as a test set since it was not present in the algorithm execution.

We collected and compared the scores obtained by some experts and non-experts to see if there exists a statistical difference between them. The users were separated in two groups A (experts) and B (non-experts). Recent answers from each group were collected and the mean of their scores were calculated. The boxplot in Figure 6.11 shows the performance of each group in this test set.

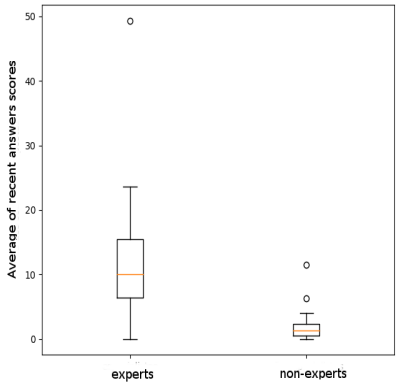


Figure 6.11: Performance of expert and non-expert users.

It can be seen that the scores of answers given by the experts are superior than the non-experts. The outlier belonging in group A refers to an user that besides having a very

high score mean also has one answer with score 262, which is much above the average. On other hand, the two outliers in group B are users with one answer with high score but the majority of their other answers have score zero. This shows that even the outliers in Group B do not have such characteristics of being experts like the ones in Group A. Both groups have users with mean equal to zero and these are users that have not answered any question in this test set or do not have obtained any positive score.

To see if there is a significant difference in the performance of the two groups, these scores were submitted to a statistic test. Through the *Kormogorov-Smirnov* we have first verified the data does not follow a normal distribution. Two hyphotheses were ellaborated: (H0 null hypothesis) e H1 (alternative hypothesis):

- **H0**: The average of the *scores* of groups A and B are equal.
- **H1**: The average of the *scores* of groups A and B are different.

The *scores* were submitted to the *Mann-Whitney* test with 95% of significance level. It was found a $p\text{-}value < 0,05$ which indicates that the *scores* achieved by the experts are bigger than the *scores* obtained by the non-experts.

Besides comparing the answer scores by experts and non-experts we have also compared the acceptance rate of each type of user.

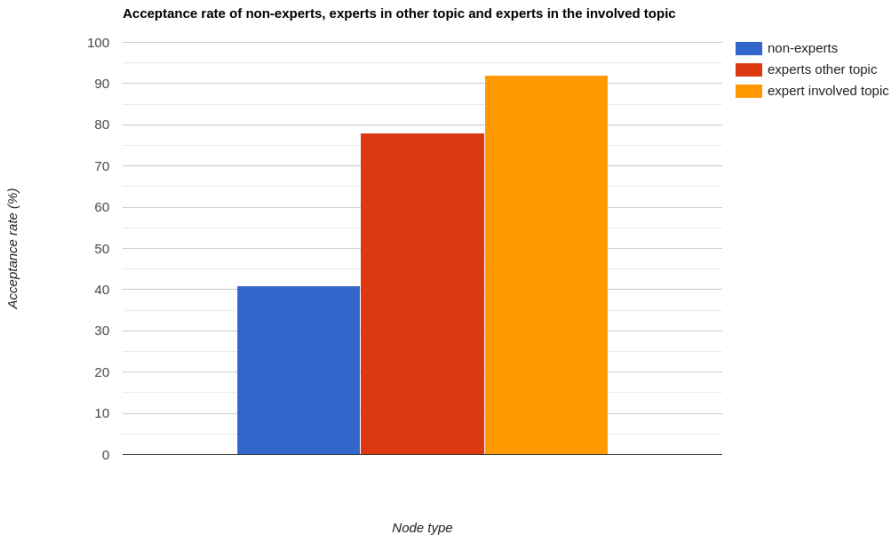


Figure 6.12: Performance of expert and non-expert users.

In Figure 6.12 three types of users were compared: non-experts, experts in any topic, experts in the question topic. This figure shows that core nodes representing experts in the question tags have the highest acceptance rate. Also, even experts in different tags have a higher acceptance rate than non-experts.

These results helps answering RQ4: *Are the core nodes developers with relevant contributions in the network, characterizing central connectors*:

The above evaluation indicates that *core* nodes detected by NetSCAN have a good performance in question-answer scenario and can represent expert developers in the network. In addition, the analysis in section 6.4 indicates shows that the *core* nodes are hubs in the star shaped network, characterizing them as central connectors.

6.5 DISCUSSION

In this Chapter we have conducted a **History Research** on a software development social network to evaluate different aspects of proposed NetO ontology and NetSCAN algorithm.

A collaborative model for the developers network was defined, with directed, weighted and heterogenous relationships. Because NetSCAN was designed for homogeneous graphs, a framework was also defined to allow the execution of the algorithm in a heterogenous graph.

Moreover, this evaluation differs from the previous Chapter in two ways. A first difference is that it performs the semantic analysis in the beginning of the process and thus semantically enrich the graph before the topological analysis, allowing it to take benefits from the enriched graph. The second difference is that it executes NetSCAN in topical and homogenous subgraphs, which allowed the automatic detection of semantic communities and their members interests.

As results, communities of software developers and expert developers were automatically detected as well as their topics of interest. To help verifying if the objectives were achieved, two research questions were stated, based on the main research questions in Chapter 1.

(i) How to discover semantic communities of developers?

To answer this question two secondary research questions were defined:

RQ1: Does the use of NetSCAN and NetO allows the detection of communities with high internal connectivity?

We have answered this question by measuring the communities cohesion using the silhouette index, as shown in the boxplot in Figure 6.8. This boxplot shows that all found communities have positive values for the silhouette index and that the average value is 0.88, which indicates high quality communities. Also, none of the communities have a negative value for the silhouette indexes, showing that all of them have better internal connectivity than external connectivity. Therefore, through this quantitative analysis we have shown that found communities have higher internal connectivity than external connectivity.

RQ2: Are the members of detected communities semantically aligned with their assigned topics of interests?

To answer this question we have extracted and analyzed the recent activity of members of the detected communities. This analysis concerned three types of people: (A) single community members; (B) specific topic community members and; (C) overlapping communities members.

After analyzing recent activity of the three types of people we have shown that most of them are still active in their community topics. It was also shown through a temporal analysis on the overlaps that the inactivity of Type C people in their topics might be related to a change in their interests. Therefore, although people of Type C might be experiencing a change of interests, it is viable to say that people are in general semantically aligned with their communities' topics of interests.

(ii) How to identify expert developers and their topics of expertise?

Again, to answer this question two other secondary research questions were elaborated.

RQ3: Are there developers who work in two or more communities simultaneously, characterizing information brokers?

The temporal analysis conducted over the overlapping communities have shown that some overlapped nodes are actively collaborating with multiple communities simultaneously, which characterizes them as information brokers. We have shown that some of them are also participating in communities of different topics, indicating that they may have multidisciplinary skills in concurrent technologies. In addition, other overlapped nodes might also indicate developers that are moving from one community to another. In this case, it might be interesting to locate these people in time in order to motivate collaborations in both communities or to facilitate the transition.

RQ4: Are the *core* nodes developers with relevant contributions in the network, characterizing central connectors?

To answer this question we have analyzed the performance of core nodes in a recent dataset, which was not used during the algorithm execution. Through the boxplot analysis in Figure 6.8 and the statistic test in Section 6.4 it was possible to see that core nodes have a performance much higher than border point nodes. Figure 6.12 also shows that the acceptance rate of core nodes are higher than border points, which is an evidence that the experts automatically detected by our approach can be recommended to give relevant answers in the Q&A context. These results indicates that core nodes have characteristics of expert developers and are central connectors in the network, answering RQ4.

7 FINAL REMARKS

This work discussed the use of complex networks, together with semantic analysis in real-world large social networks. We dealt with the problem of identifying people who could help maintain the network connectivity. Therefore, the main objective of this work was to detect central connectors, information brokers and social communities in an attempt to identify influential people in social networks. In addition, we aimed to identify the semantic context of communities and their members in order to motivate connections between people that have similar interests, even though they are not explicit linked.

We developed a proposal that found central connectors and information brokers with the aim of identifying influential people in social networks so as to connect people with similar interests, even if such interests are not explicit. We have first proposed a novel density-based algorithm for automatic detection of overlapping communities and important nodes in social networks, called NetSCAN. Then, an ontology called NetO was proposed to identify topics of interests of actors in social networks and to detect people with similar interests, thus helping semantic analysis in social networks.

Our proposal was first tested separately in controlled experiments with well known datasets, showing that it achieves the expected behaviour and that results are compatible with the ones found in the literature. After these preliminary tests we have conducted two history research evaluations using NetSCAN and NetO together perform structural and semantic analysis over two real-world social networks.

In a first evaluation, a scientific social network was modeled and analyzed based on data from DBLP. The second evaluation analyzed a software development social network, based on StackOverflow data. Through extensive analysis on these social networks, we answered the two research questions: (i) How to identify nodes who help maintain the network connectivity, dis- seminating information and linking groups/subgroups? (ii) How to discover semantic connections between nodes, also including their connections based on their context, even though such connections are not explicit?

By executing NetSCAN on DBLP, many scientific communities and influential researchers were found. The semantic analysis showed that researchers in the same community had in fact similar context and interests, which points to the quality of the results

generated by the algorithm. As a result, we could check the quality of the context information using both the topological and the semantic analyses.

In addition, the proposal extracted the semantic meaning from distributed scientific repositories: DBLP, digital scientific libraries and ACM domain taxonomy (CCS). These scientific repositories were accessed through their APIs and processed using context data, delivering information for helping in the decision making of communities and researchers.

The experiments in the software development network have also helped answering the research questions. By executing the semantic enrichment with NetO support before NetSCAN we have detected communities and their topics of interests automatically. It was also possible to find that core nodes may represent multidisciplinary and expert developers, characterizing core nodes as information brokers and central connectors. The analyses point to the feasibility of both topological and semantic analyses using the NetSCAN algorithm and NetO ontology. Based on the evaluation performed, it can be outlined that there are indications that the achieved result was satisfactory.

As NetSCAN is based on DBSCAN it also has some limitations regarding the parameters selection. The choosing of eps and minPts relies on the domain knowledge and different choices for these parameters might produce different results.

As further work, we aim to devise a strategy based on graph metrics to indicate the optimal parameterization in the refinement of the groups, since the computational effort in this step can increase considerably. We also plan to tackle some of the applications that can benefit from the discovered communities and core nodes, such as expert recommendation and link prediction.

REFERENCES

AGGARWAL, C. C. (Ed.). **Social Network Data Analytics**, 2011. Disponível em: <https://doi.org/10.1007/978-1-4419-8462-3>.

BHAT, S. Y.; ABULAISH, M. A density-based approach for mining overlapping communities from social network interactions. In: **Proceedings of the 2Nd International Conference on Web Intelligence, Mining and Semantics**, 2012. (WIMS '12), p. 9:1–9:7. ISBN 978-1-4503-0915-8. Disponível em: <http://doi.acm.org/10.1145/2254129.2254143>.

CROSS, R.; PARKER, A. **The Hidden Power of Social Networks: Understanding how Work Really Gets Done in Organizations**, 2004. ISBN 9781591392705. Disponível em: <https://books.google.com.br/books?id=vQ3mM4Vpix8C>.

DAS, K.; SAMANTA, S.; PAL, M. Study on centrality measures in social networks: a survey. **Social Network Analysis and Mining**, Springer Nature, v. 8, n. 1, feb 2018. Disponível em: <https://doi.org/10.1007/s13278-018-0493-2>.

DERÉNYI, I.; PALLA, G.; VICSEK, T. Clique percolation in random networks. **Physical Review Letters**, jan. 2005. Disponível em: <http://link.aps.org/doi/10.1103/PhysRevLett.94.160202>.

ERETEO, G.; BUFFA, M.; GANDON, F.; GROHAN, P.; LEITZELMAN, M.; SANDER, P. A state of the art on social network analysis and its applications on a semantic web. 01 2008.

ESTER, M.; KRIEGEL, H.-P.; SANDER, J.; XU, X. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: **Proceedings of the Second International Conference on Knowledge Discovery and Data Mining**, 1996. (KDD'96), p. 226–231. Disponível em: <http://dl.acm.org/citation.cfm?id=3001460.3001507>.

FALKOWSKI, T.; BARTH, A.; SPILIOPOULOU, M. Dengraph: A density-based community detection algorithm. In: **In Proc. of the 2007 IEEE / WIC / ACM In-**

ternational Conference on Web Intelligence,, 2007. p. 112–115. Disponível em: <http://wwwiti.cs.uni-magdeburg.de/ tfalkows/publ/2007/WI$_F alBarSpi$07.*pdf*>.

FORTUNATO, S. Community detection in graphs. **Physics Reports**, v. 486, n. 3, p. 75 – 174, 2010. ISSN 0370-1573. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0370157309002841>.

GIALAMPOUKIDIS, I.; TSIKRIKA, T.; VROCHIDIS, S.; KOMPATSIARIS, I. Community detection in complex networks based on dbscan* and a martingale process. In: **2016 11th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP)**, 2016. p. 1–6.

GRABOWICZ, P. A.; RAMASCO, J. J.; MORO, E.; PUJOL, J. M.; EGUILUZ, V. M. Social features of online networks: The strength of intermediary ties in online social media. **PLoS ONE**, Public Library of Science (PLoS), v. 7, n. 1, p. e29358, jan 2012. Disponível em: <https://doi.org/10.1371/journal.pone.0029358>.

GUILLE, A. Information diffusion in online social networks. In: **Proceedings of the 2013 Sigmod/PODS Ph.D. symposium on PhD symposium - SIGMOD'13 PhD Symposium**, 2013. Disponível em: <https://doi.org/10.1145/2483574.2483575>.

HAN, J.; KAMBER, M.; PEI, J. **Data Mining: Concepts and Techniques**. 3rd. ed., 2011. ISBN 0123814790, 9780123814791.

HLAOUI, A.; WANG, S. A direct approach to graph clustering. In: **Neural Networks and Computational Intelligence**, 2004. p. 158–163. Disponível em: <http://dblp.uni-trier.de/db/conf/nci/nci2004.htmlHlaouiW04>.

HORTA, V.; STRÖELE, V.; CAMPOS, F.; DAVID, J. M. N.; BRAGA, R. M. M. Redes sociais científicas: análise topológica da influência dos pesquisadores. In: **XXXII Simpósio Brasileiro de Banco de Dados - Short Papers, Uberlandia, MG, Brazil, October 4-7, 2017.**, 2017. p. 282–287. Disponível em: <http://sbbd.org.br/2017/wp-content/uploads/sites/3/2018/02/p282-287.pdf>.

HORTA, V.; STRÖELE, V.; OLIVEIRA, J.; BRAGA, R. M. M.; DAVID, J. M. N.; CAMPOS, F. Análise de colaboração em desenvolvimento global de software. In: **33rd Annual Brazilian Symposium on Databases, SBBD 2018,**

Rio de Janeiro, RJ, Brazil, August 25-26, 2018., 2018. p. 145–156. Disponível em: <http://sbbd.org.br/2018/wp-content/uploads/sites/5/2018/08/145-sbbd_2018-fp.pdf>.

HORTA, V.; STRöELE, V.; BRAGA, R.; DAVID, J. M. N.; CAMPOS, F. Analyzing scientific context of researchers and communities by using complex network and semantic technologies. **Future Generation Computer Systems**, v. 89, p. 584 – 605, 2018. ISSN 0167-739X. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0167739X17328431>.

HU, Y.; WANG, S.; REN, Y.; CHOO, K.-K. R. User influence analysis for github developer social networks. **Expert Systems with Applications**, Elsevier BV, v. 108, p. 108–118, oct 2018. Disponível em: <https://doi.org/10.1016/j.eswa.2018.05.002>.

JAYAWEERA, N.; PERERA, K.; MUNASINGHE, J. Centrality measures to identify traffic congestion on road networks: A case study of sri lanka. **IOSR Joournal of Mathematics**, v. 13, p. 13–19, 04 2017.

KIANIAN, S.; KHAYYAMBASHI, M. R.; MOVAHHEDINIA, N. Fuseo: Fuzzy semantic overlapping community detection. **Journal of Intelligent Fuzzy Systems**, IOS Press, v. 32, n. 6, p. 3987–3998, May 2017. ISSN 1064-1246. Disponível em: <http://doi.org/10.3233/JIFS-151276>.

KITCHENHAM, B. A. **Kitchenham, B.: Guidelines for performing Systematic Literature Reviews in software engineering. EBSE Technical Report EBSE-2007-01**, 2007.

KOSCHÜTZKI, D.; SCHREIBER, F. Centrality analysis methods for biological networks and their application to gene regulatory networks. In: **Gene regulation and systems biology**, 2008.

LESKOVEC, J.; LANG, K. J.; MAHONEY, M. Empirical comparison of algorithms for network community detection. In: **Proceedings of the 19th International Conference on World Wide Web**, 2010. (WWW '10), p. 631–640. ISBN 978-1-60558-799-8. Disponível em: <http://doi.acm.org/10.1145/1772690.1772755>.

LI, X.; TAN, Y.; ZHANG, Z.; TONG, Q. Community detection in large social networks based on relationship density. In: **2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)**, 2016. Disponível em: <https://doi.org/10.1109

LIU, P.; RAAHEMI, B.; BENYOUCEF, M. Knowledge sharing in dynamic virtual enterprises: A socio-technological perspective. **Knowledge-Based Systems**, v. 24, n. 3, p. 427 – 443, 2011. ISSN 0950-7051. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0950705110001760>.

LUO, T.; ZHONG, C.; YING, X.; FU, J. Detecting community structure based on edge betweenness. In: **2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)**, 2011. Disponível em: <https://doi.org/10.1109

MAMYKINA, L.; MANOIM, B.; MITTAL, M.; HRIPCSAK, G.; HARTMANN, B. Design lessons from the fastest q&a site in the west. In: **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**, 2011. (CHI '11), p. 2857–2866. ISBN 978-1-4503-0228-9. Disponível em: <http://doi.acm.org/10.1145/1978942.1979366>.

MEENA, J.; DEVI, V. S. Overlapping community detection in social network using disjoint community detection. In: **2015 IEEE Symposium Series on Computational Intelligence**, 2015. Disponível em: <https://doi.org/10.1109/ssci.2015.114>.

MENG, Z.; GANDON, F.; FARON-ZUCKER, C. Qasm: a qa social media system based on social semantics. 10 2014.

MENG, Z.; GANDON, F.; ZUCKER, C. F. Simplified detection and labeling of overlapping communities of interest in question-and-answer sites. In: **2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)**, 2015. v. 1, p. 107–114.

MENG, Z.; GANDON, F.; ZUCKER, C. F. Joint model of topics, expertises, activities and trends for question answering web applications. In: **2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)**, 2016. p. 296–303.

MILGRAM, S. The small-world problem. **Psychology Today**, v. 1, n. 1, 1967.

NEWMAN, M. E. J.; GIRVAN, M. Finding and evaluating community structure in networks. **Physical Review E**, American Physical Society (APS), v. 69, n. 2, feb 2004. Disponível em: <https://doi.org/10.1103

PAPADOPOULOS, S.; KOMPATSIARIS, Y.; VAKALI, A.; SPYRIDONOS, P. Community detection in social media. **Data Min. Knowl. Discov.**, Kluwer Academic Publishers, Hingham, MA, USA, v. 24, n. 3, p. 515–554, maio 2012. ISSN 1384-5810. Disponível em: <http://dx.doi.org/10.1007/s10618-011-0224-z>.

Puig-Centelles, Anna and Ripolles, Oscar and Chover, Miguel. Surveying the identification of communities. **Int. J. Web Based Communities**, Inderscience Publishers, Inderscience Publishers, Geneva, SWITZERLAND, v. 4, n. 3, p. 334–347, jul. 2008. ISSN 1477-8394. Disponível em: <http://dx.doi.org/10.1504/IJWBC.2008.019193>.

RÍOS, S. A.; MUÑOZ, R. Content patterns in topic-based overlapping communities. **The Scientific World Journal**, Hindawi Limited, v. 2014, p. 1–11, 2014. Disponível em: <https://doi.org/10.1155/2014/105428>.

STRÖELE, V.; CRIVANO, R.; ZIMBRÃO, G.; SOUZA, J. M.; CAMPOS, F.; DAVID, J. M. N.; BRAGA, R. Rational erdös number and maximum flow as measurement models for scientific social network analysis. **Journal of the Brazilian Computer Society**, v. 24, n. 1, p. 6, Jul 2018. ISSN 1678-4804. Disponível em: <https://doi.org/10.1186/s13173-018-0070-6>.

STRöELE, V.; CAMPOS, F.; PEREIRA, C. K.; ZIMBRãO, G.; SOUZA, J. M. Information extraction to improve link prediction in scientific social networks. In: **2016 IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD)**, 2016. p. 515–520.

TAN, P.-N.; STEINBACH, M.; KUMAR, V. **Introduction to Data Mining, (First Edition)**, 2005. ISBN 0321321367.

WANG WENZHONG TANG, B. S. J. F. C.; WANG, Y. Review on community detection algorithms in social networks. In: **2015 IEEE International Conference on Progress in Informatics and Computing (PIC)**, 2015. p. 551–555.

WASSERMAN, S.; FAUST, K. **Social network analysis: Methods and applications**, 1994. Disponível em: <http://scholar.google.com/scholar.bib?q=info:gET6m8icitMJ:scholar.google.com/output=citation $0,5as_vis = 1ct = citationcd = 0$>.

WU, F.; HUBERMAN, B. A.; ADAMIC, L. A.; TYLER, J. R. Information flow in social groups. **Physica A: Statistical Mechanics and its Applications**, v. 337, n. 1, p. 327 – 335, 2004. ISSN 0378-4371. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0378437104000548>.

XIE, J.; KELLEY, S.; SZYMANSKI, B. K. Overlapping community detection in networks: The state-of-the-art and comparative study. **ACM Comput. Surv.**, ACM, New York, NY, USA, v. 45, n. 4, p. 43:1–43:35, ago. 2013. ISSN 0360-0300. Disponível em: <http://doi.acm.org/10.1145/2501654.2501657>.

YANG, C.; MA, J.; SILVA, T.; LIU, X.; HUA, Z. A multilevel information mining approach for expert recommendation in online scientific communities. 05 2014.

YANG, J.; LESKOVEC, J. Defining and evaluating network communities based on ground-truth. In: **2012 IEEE 12th International Conference on Data Mining**, 2012. p. 745–754. ISSN 1550-4786.

YIN, R. K. **Case Study Research: Design and Methods (Applied Social Research Methods)**. Fourth edition., 2008. ISBN 1412960991. Disponível em: <http://www.amazon.de/Case-Study-Research-Methods-Applied/dp/1412960991

YUAN, W.; GUAN, D.; LEE, Y.-K.; LEE, S.; HUR, S. J. Improved trust-aware recommender system using small-worldness of trust networks. **Know.-Based Syst.**, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 23, n. 3, p. 232–238, abr. 2010. ISSN 0950-7051. Disponível em: <http://dx.doi.org/10.1016/j.knosys.2009.12.004>.

ZACHARY, W. An information flow model for conflict and fission in small groups1. v. 33, 11 1976.