Relationship discovery in social networks

Elie Raad

UNIVERSITÉ DE BOURGOGNE

UFR Sciences et Techniques

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE BOURGOGNE

Discipline : **Informatique**

par

Elie RAAD

Relationship discovery in social networks (Découverte des relations dans les réseaux sociaux)

Le 22 décembre 2011

Devant le jury composé de

Ernesto DAMIANI	Professeur, Université de Milan	Rapporteur
Nhan LE THANH	Professeur, Université de Nice - Sophia Antipolis	Rapporteur
Ahmed MOSTEFAOUI	Maître de Conférences - HDR, Université de Franche Comté	Examinateur
Richard CHBEIR	Maître de Conférences - HDR, Université de Bourgogne	Co-directeur
Albert DIPANDA	Professeur, Université de Bourgogne	Co-directeur

À mon pays, le Liban,

À mes parents,

À Mylène.

Acknowledgements

I would like to express my sincere gratitude and humble acknowledgement to all the people who have contributed to the preparation and completion of this research.

Foremost, no words can express my gratefulness to my advisors, Dr. Richard Chbeir and Prof. Albert Dipanda, who have expertly guided me through this thesis. Their support and guidance were key players in making this research feasible.

It is with great gratitude that I thank Dr. Richard Chbeir for his constant support and encouragement. His valuable comments and suggestions were a big inspiration throughout each step of the process and helped me overcome many obstacles. I am fortunate and honored for working with Dr. Chbeir as he is not only a great mentor, but also a role model and more importantly, a friend.

I would like to thank Prof. Albert Dipanda for his confidence in me and for giving me the opportunity to be his student and work on this research within the scope of LE2I laboratory, University of Bourgogne.

I am extremely grateful to Prof. Ernesto Damiani and Prof. Nhan Le Thanh for examining my thesis report. Their valuable reviews have resulted in a further improvement of the work. I would also like to thank Dr. Ahmed Mostefaoui for being part of the PhD examination committee of my thesis.

Among my colleagues at LE2I, I would like to acknowledge the support of Fekade Getahun, Bechara Al Bouna, Mônica Ribeiro Porto Ferreira, Elie Abi Lahoud, Georges Chalhoub, Joe Tekli, and Gilbert Tekli. I would like also to thank Aries Muslim, Guillermo Valente Gomez Caprio, Raji Ghawi, Sylvain Rakotomalala, and Damien Leprovost for their help and sympathy.

A special thank goes to my fiancée Mylène, for her love, encouragement, and understanding. I am also extremely grateful for the support I have received from her family. Last but not least, my deepest gratitude goes to my parents, my sister Eliana, and my brother Charbel, for all the love and support they give me every day.

Remerciements

C'est avec un immense plaisir que je tiens à remercier très sincèrement toutes les personnes qui m'ont aidé et qui ont ainsi contribué à la réalisation de ce travail.

Je tiens à exprimer ici toute l'estime et l'admiration que je porte au docteur Richard Chbeir et au professeur Albert Dipanda qui m'ont guidé tout au long de ma recherche. Ce travail n'aurait jamais pu aboutir sans leurs aides et leurs conseils.
J'aimerais plus particulièrement exprimer ma reconnaissance au docteur Richard Chbeir pour son soutien et ses encouragements tout au long de ce travail. Ses commentaires et ses indications m'ont aidé à surmonter les différents problèmes rencontrés ces dernières années. Plus qu'un mentor, il est devenu un ami et je suis conscient de la chance et de l'honneur que j'ai eu de pouvoir travailler avec lui.
Je voudrais remercier le professeur Albert Dipanda pour sa confiance et pour m'avoir accepté en tant qu'étudiant et m'avoir ainsi donné l'opportunité d'effectuer ce travail de recherche au sein du laboratoire LE2I de l'Université de Bourgogne.

J'adresse mes sincères remerciements au professeur Ernesto Damiani et au profresseur Nhan Le Thanh pour avoir consacré de leur temps pour rapporter mon travail de thèse. Je remercie également le docteur Ahmed Mostefaoui d'avoir bien voulu faire partie des membres de mon jury.

Je voudrais également remercier mes collègues du laboratoire LE2I pour leur soutien, et plus particulièrement : Fekade Getahum, Bechara Al Bouna, Mônica Ribeiro Porto Ferreira, Elie Abi Lahoud, Georges Chalhoub, Joe Telki, Gilbert Tekli. Mes remerciements vont aussi à Aries Muslim, Guillermo Valente Gomez Caprio, Raji Ghawi, Sylvain Rakotomalala et Damien Leprovost pour leur aide et leur sympathie.

Je souhaite exprimer ma gratitude envers Mylène, ma fiancée, pour son amour, ses encouragements et sa compréhension. Également, je suis extrêmement reconnaissant envers sa famille pour son soutien. Enfin, je tiens à remercier mes parents, ma soeur Eliana et mon frère Charbel pour l'amour et le soutien qu'ils m'ont donnés chaque jour.

Abstract

In recent years, social network sites exploded in popularity and become an important part of the online activities on the web. This success is related to the various services/functionalities provided by each site (ranging from media sharing, tagging, blogging, and mainly to online social networking) pushing users to subscribe to several sites and consequently to create several social networks for different purposes and contexts (professional, private, etc.). Nevertheless, current tools and sites provide limited functionalities to organize and identify relationship types within and across social networks which is required in several scenarios such as enforcing users' privacy, and enhancing targeted social content sharing, etc. Particularly, none of the existing social network sites provides a way to automatically identify relationship types while considering users' personal information and published data. In this work, we propose a new approach to identify relationship types among users within either a single or several social networks. We provide a user-oriented framework able to consider several features and shared data available in user profiles (e.g., name, age, interests, photos, etc.). This framework is built on a rule-based approach that operates at two levels of granularity: 1) within a single social network to discover *social relationships* (i.e., colleagues, relatives, friends, etc.) by exploiting mainly photos' features and their embedded metadata, and 2) across different social networks to discover *co-referent relationships* (same real-world persons) by considering all profiles' attributes weighted by the user profile and social network contents. At each level of granularity, we generate a set of basic and derived rules that are both used to discover relationship types. To generate basic rules, we propose two distinct methodologies. On one hand, social relationship basic rules are generated from a photo dataset constructed using *crowdsourcing*. On the other hand, using all weighted attributes, co-referent relationship basic rules are generated from the available pairs of profiles having the same unique identifier(s) attribute(s) values. To generate the derived rules, we use a mining technique that takes into account the context of users, namely by identifying frequently used valid basic rules for each user. We present here our prototype, called *RelTypeFinder*, implemented to validate our approach. It allows to discover appropriately different relationship types, generate synthetic datesets, collect web data and photo, and generate mining rules. We also describe here the sets of experiments conducted on real-world and synthetic datasets. The evaluation results demonstrate the efficiency of the proposed relationship discovery approach.

Keywords: Relationship discovery, Social networks, Rule-based relationship identification, Co-referent users, Link mining, Link type prediction, Entity resolution, Classification, Crowdsourcing, User profiles, Photos, Metadata.

Résumé

Les réseaux sociaux occupent une place de plus en plus importante dans notre vie quotidienne et représentent une part considérable des activités sur le web. Ce succès s'explique par la diversité des services/fonctionnalités de chaque site (partage des données souvent multimédias, tagging, blogging, suggestion de contacts, etc.) incitant les utilisateurs à s'inscrire sur différents sites et ainsi à créer plusieurs réseaux sociaux pour diverses raisons (professionnelle, privée, etc.). Cependant, les outils et les sites existants proposent des fonctionnalités limitées pour identifier et organiser les types de relations ne permettant pas de, entre autres, garantir la confidentialité des utilisateurs et fournir un partage plus fin des données. Particulièrement, aucun site actuel ne propose une solution permettant d'identifier automatiquement les types de relations en tenant compte de toutes les données personnelles et/ou celles publiées. Dans cette étude, nous proposons une nouvelle approche permettant d'identifier les types de relations à travers un ou plusieurs réseaux sociaux. Notre approche est basée sur un framework orienté-utilisateur qui utilise plusieurs attributs du profil utilisateur (nom, age, adresse, photos, etc.). Pour cela, nous utilisons des règles qui s'appliquent à deux niveaux de granularité : 1) au sein d'un même réseau social pour déterminer les *relations sociales* (collègues, parents, amis, etc.) en exploitant principalement les caractéristiques des photos et leurs métadonnées, et, 2) à travers différents réseaux sociaux pour déterminer les *utilisateurs co-référents* (même personne sur plusieurs réseaux sociaux) en étant capable de considérer tous les attributs du profil auxquels des poids sont associés selon le profil de l'utilisateur et le contenu du réseau social. À chaque niveau de granularité, nous appliquons des règles de base et des règles dérivées pour identifier différents types de relations. Nous mettons en avant deux méthodologies distinctes pour générer les règles de base. Pour les relations sociales, les règles de base sont créées à partir d'un jeu de données de photos créées en utilisant le *crowdsourcing*. Pour les relations de co-référents, en utilisant tous les attributs, les règles de base sont générées à partir des paires de profils ayant des identifiants de mêmes valeurs. Quant aux règles dérivées, nous utilisons une technique de fouille de données qui prend en compte le contexte de chaque utilisateur en identifiant les règles de base fréquemment utilisées. Nous présentons notre prototype, intitulé *RelTypeFinder*, que nous avons implémenté afin de valider notre approche. Ce prototype permet de découvrir différents types de relations, générer des jeux de données synthétiques, collecter des données du web, et de générer les règles d'extraction. Nous décrivons les expériementations que nous avons menées sur des jeux de données réelles et syntéthiques. Les résultats montrent l'efficacité de notre approche à découvrir les types de relations.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

"We cannot become what we need to be by remaining what we are." - Max DePree

The last decade has witnessed the rapid development of several web-based collaboration tools and communication technologies calling on the participation of a large number of users. These novel forms of communication and collaboration shed the light on the potential of leveraging the web to solve various types of challenges by recruiting a large number of users: 1) web users volunteered to collaboratively write encyclopedia articles and thus participated in the advent of Wikipedia[1], 2) the wisdom of crowd used mainly to review and evaluate products on online stores such as Amazon[2], 3) keyword tags used to bookmark and categorize web pages on social web systems such as del.ici.ous[3], etc.

Nowadays, web users are highly interested in joining and using social network sites (e.g., Facebook[4], LinkedIn[5], Google+[6]). These popular sites become an important part of the online activities on the web and one of the most influencing media. Information available on social networks commonly describes persons and their interactions, along with their published data. While the number of social network users is rapidly growing, there is an increasing need to deal with a wide range of challenges such as facilitating communication among users, building personal content management systems for better data organization, storage, and retrieval, analyzing social interactions for better relationship management, etc.

Appropriate relationship management, which refers to maintaining information about a user's contacts, is of particular importance on social network sites since current social network

[1] http://www.wikipedia.org/
[2] http://www.amazon.com/
[3] http://delicious.com/
[4] http://www.facebook.com/
[5] http://www.linkedin.com/
[6] https://plus.google.com/

sites fall short of providing relevant related (relationship management) functionalities.

1. **Within a single social network:** One of the most fundamental relationship management tasks aims at identifying social relationship types that are relevant between a main user and her contacts within the same social network site. While myriad social networks' services assist users to find new contacts and establish new connections (e.g., friend suggestion systems through locations [2], based on interactions [3], etc.), users get connected to different types of contacts such as friends, relatives, and colleagues. Nevertheless, social relationship types between a main user and her contacts are rarely identified neither by the user nor by the application within existing social network sites [3] [4]. In fact, labeling and updating (manually) the relationship types between users and their contacts is a complex, time-consuming, and tedious task that requires constant maintenance [5] [6]. Despite that some social network sites provide users with the possibility to manually identify the type of relationship with their contacts, this option is skipped most of the time [7]. Consequently, only the presence of a relationship is indicated. Furthermore, current relationship management tools (such as alphabetized lists, Google+ circles, grids of photos on Facebook, etc.) cannot organize contacts automatically into relationship-based groups [8]. It is therefore important to provide efficient means to automatically discover/identify social relationship types between a main user and her contacts within the same social network site.

2. **Across different social network sites:** Social network users increasingly interact with each other using a wide variety of communication tools offered by different social network sites. In reality, each social network site offers particular services and functionalities to target a well-defined community in the real world. To make use of the provided functionalities and to stay tuned with their related members, users create several accounts on various sites where they disclose personal and professional information. These information are stored within their user profile(s) which consists of different attributes, such as name, age, interests, educational history, photos, etc. A challenging task is whether one can identify profiles that refer to the same real-world person (co-referent users) across different social network sites. Identifying co-referent users can be regarded as an additional relationship type between users. Obviously, multiple and different representations of the same real-world person across different social network sites are one of the most intriguing problems. This has contributed to the emergence of new users' related needs to perform various inter-networks' operations (e.g., finding the intersection or the difference of two sets of users between two social networks).

Understanding social interactions arising within social network sites is a challenge that is

related to the behavior of social network users, the social data that is being shared, and the appropriate tools and services available which allow user interactions to grow over time. In this work, we are primarily interested in studying relationships within a single or across different social network sites. This requires adaptable relationship management where links between social network users are semantically enriched with corresponding types over social network sites.

1.1 Motivation

In this section, we outline the main motivations behind our interest in discovering/identifying relationship types between users within the same social network or across different social network sites and explain the advantages of identifying corresponding relationships in the light of different situations.

1.1.1 Current Use of Social Network Sites

According to an In-Stat's survey[7], out of more than 10 billion registered accounts in 2010, the active social networking and online worlds (SNOW) accounts amount to nearly 4.5 billion. Another study published by Nielsen on social networking [1] shows that social network sites and blogs dominate Americans' time online and now account for nearly a quarter of the time spent online. As shown in Figure 1.1, with 53.5 billion minutes of time spent online, users spent more time on Facebook than on Google[8], Yahoo[9], Youtube[10], Microsoft[11], Wikipedia, and Amazon combined.

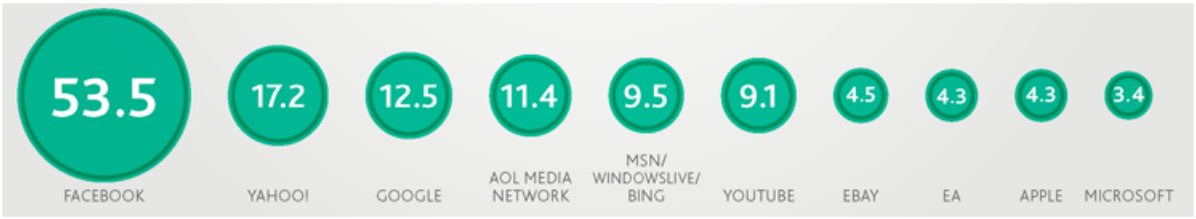


Figure 1.1: Total time in billions of minutes spent online by U.S. users on the top 10 web brands. Source: Nielsen Report [1].

[7] http://www.instat.com/press.asp?ID=3085&sku=IN1004659CM
[8] http://www.google.com/
[9] http://www.yahoo.com/
[10] http://www.youtube.com/
[11] http://www.microsoft.com/

Social networks currently allow users to exchange messages, publish photos, post comments, etc. With more than 250 million photos on average uploaded every day on Facebook[12], such social network sites will indisputably shape the future of personal photography. Needless to say that photos are one of the most important means that capture a variety of information including social ties, family relations, meaningful locations, feelings and preferences of persons, important events, etc. [9].

As photos on social networks are gaining more importance, they are of valuable consideration for social network users. Each of these published photos can be assigned intentionally, for example, by a textual description or a comment, as well as extensionally by tagging[13] users that are relevant, in one way or another, in the published photo. Consequently, the online presence of social network users goes beyond their published data; it also concerns data in which they are tagged, but owned by others.

Photos are embedded in our daily lives; they capture natural and day-to-day human interactions and hold instantly available technical information (location, date, and time of photo capture). Moreover, they can be easily shared, and users can add their descriptions and comments. Consequently, photos can easily shape our social activities and relationships. The current situation related to the use of photos can be summarized by the following blog post statement[14]: *"Our parents took photos to try to hold on to the past. We take photos to create the present"*.

1.1.2 Identifying Social Relationship Types

Today's social network users can be connected to different types of contacts such as colleagues, relatives, friends, etc. In fact, contacts on social networks and real-life contacts are increasingly interwoven. Relationship discovery has found considerable interest recently due to the growing number of social network users and the increasing need to analyze social interactions.

As users maintain large lists of contacts on social network sites [10], one might rightly ask how many of these contacts correspond to real-life contacts, and how many are real friends, relatives, colleagues, or strangers? Figure 1.2 represents a main user connected with her contacts with identified social relationship types. In fact, identifying social relationship types is useful in many situations such as:

1. **Managing contact lists:** as users get connected to a big number of contacts, manag-

[12]http://www.facebook.com/press/info.php?statistics - October 2011

[13]In fact, on social networks, the concept of *tagging* refers mainly to the task of linking a published item to one or several contacts.

[14]http://cloudhead.headmine.net/post/273997836/realtime

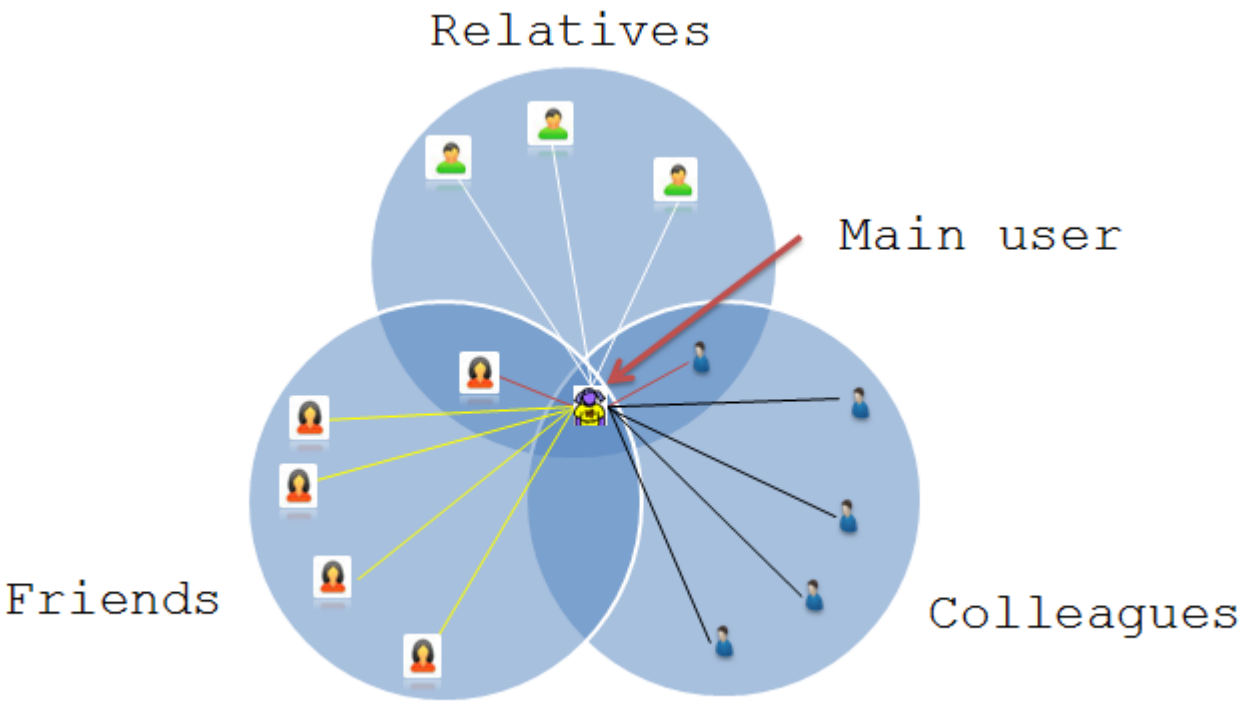


Figure 1.2: A main user having her contacts grouped based on the identified social relationship type.

ing large contact lists is a tedious task. One potential solution to this challenge lies in automatically identifying the social relationship type with each contact. In fact, organizing contacts based on the relevant relationship types can be useful for different situations where targeted social content sharing and filtering are needed.

2. **Improve sharing:** although current social networks let users manage and control their privacy settings, they typically do so in terms of contact identities or grouped lists of contacts. Controlling access to own resources is rather driven by relationships that social network users share with their contacts such as colleagues, relatives, friends, etc. Treating all their contacts in the same way, without differentiating one user from another, is an unsafe and restrictive practice [11]. For instance, a user might want to prevent her relatives from viewing content posted by her friends or need to share some data and interact with some but not with all of the members of her social network.

3. **Facilitate tagging:** since relationship management involves the task of retrieving information related to contacts, it is crucial to provide users with the possibility to list,

visualize, and group photos based on different criteria. Indeed, automatically identifying the type of relationship enables a user to visualize the set of photos where she is tagged so she can decide to remove/add a tag, or a comment, block/add a contact, etc. For instance, a main user might want to visualize the photos where she is tagged by one or several colleagues, relatives, or friends. However, inspecting each photo and verifying manually the relationship type of contact who tagged the main user is hardly possible.

4. **Privacy Improvement:** privacy protection is one major concern for social network users. Here, we develop two main privacy-related points:

   (a) **Protecting some relationships:** as a common practice, social network users upload and tag photos taken at different events. Photos from a professional event, for instance, can be uploaded and made public on any social network site. For personal reasons, a user, tagged with her husband, might request to remove all the tags from photos where she appears with him. Removing/hiding tags from photos is certainly one way to hide photos from being displayed on the user profile and thus help preserve relationship privacy.

   (b) **Preventing exchanges between contacts of different relationships:** contacts of different relationship types can interact with each other by posting comments and interacting over the main user's profile (such as on the user profile's wall, photos, posts, etc.). For instance, a user might want to prevent her relatives from connecting, posting comments, and communicating with her colleagues hence having the type of relationship of all contacts can help restrict potential interactions between different types of contacts.

In essence, one of the key challenges which arises in the context of social networks is relationship management. Social networks' users are committed to addressing the multifaceted issues related to relationship management. At any time, a user can add a new contact to her profile and interact with different contacts. Relationship management is a challenge that is increasingly rising on social networks due to the multitude of contacts, the social relationship types that evolve over time, and the big amount of data published and tagged by users. Consequently, enriching links between users with appropriate types is a key challenge for better relationship management.

1.1.3 Identifying Co-referent Relationship Types

With the rise of social network sites, the number of services provided to end users is significantly increasing. These services encompass a variety of communication tools for connecting users together, sharing information, and collaborative tagging. To make use of the provided services and functionalities and to keep being tuned with its related members, users create several accounts on various sites. Recently, two released statistics on social network users attract the attention:

- With the huge number of social network users, nowadays 4 active internet users out of 5 are currently social network users, according to the Nielsen study [1]
- Over half of the social network users (51%) have two or more online profiles[15], with the majority of these users (83%) having their profiles on different social network sites for professional and personal reasons.

Together, these two statistics can be seen as a motivation toward addressing the challenge of identifying co-referent users, also known as user profile matching on social networks, as illustrated in Figure 1.3.

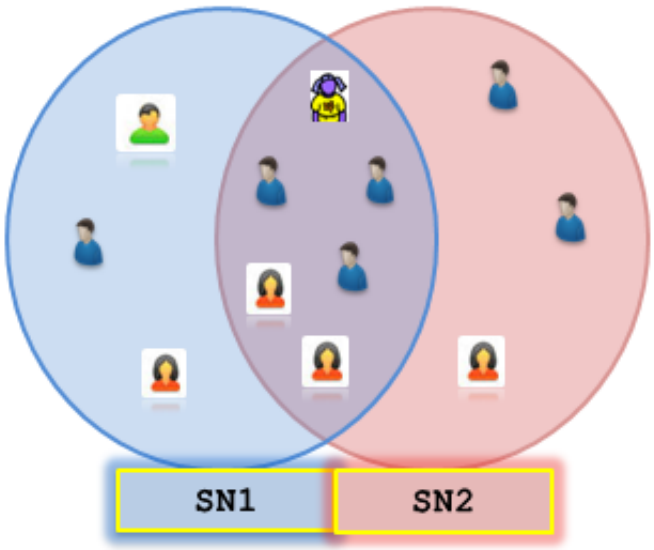


Figure 1.3: A representation of a main user having different profiles across two social network sites (SN1 and SN2) and where co-referent contacts are represented as the intersection between these two social network sites.

[15] http://www.pewinternet.org/Reports/2009/Adults-and-Social-Network-Websites.aspx

This is useful in many situations such as:

1. **Aggregating data of co-referent users:** social network users publish different information on several social network sites either purposely or accidently. Each profile, aside from being created on different sites, may only contain a portion of the complete profile information of the user. Aggregating data of co-referent users can be regarded as the task of combining and merging information in order to automatically create a complete representation of a user profile from her disparate profiles on different social sites.
2. **Linking co-referent users:** this is seen as an alternative to aggregating co-referent users. The aim is to link co-referent users without merging the data of both profiles. Actually, linking co-referent users is used to group the updates and synchronize the notifications related to a pair of co-referent users after their discovery.

Finding common users is one of the interesting challenges on social networks. It is quite useful for users who want to verify if one (or several) of their contacts on one social network site also exists in another social network site. Currently, users manually search for their contacts by trying to find profiles having the same or similar names on both sites. Automating such task and considering a wider set of profile attributes is highly desirable for matching profiles of co-referent users.

To sum up, the objective of this work is to leverage social network profiles and their related data for discovering relationship types between social network users, which is a necessary step toward enriching links within a single or across several social network sites.

1.2 Contributions

To tackle the relationship type discovery challenges, we propose a rule-based approach as a way to semantically enrich relationships among social network users. The contributions of this work focus on providing the ability to discover relationship between users within one or across several social network sites. The main contributions of this work are summarized as follows:

- **User-based framework:** with the huge amount of information available online, different social network elements (user profiles, photo albums, photos, etc.) are involved in the daily activities and interactions between users. This framework is able to identify the relationship types by taking into account all these available information, along with the preferences of the main user.

- **Rule-based approach:** we propose a set of dedicated rules which includes basic and derived rules:

  1. Basic rules: are considered to be the most trustworthy.
  2. Derived rules: these rules are automatically generated using a mining technique.

  Basic and derived rules can be used as intra-social rules to identify the appropriate social relationship types between a main user and her contacts. They can be also used as inter-social rules to identify the same real-world contacts of a main user having a user profile on two distinct social network sites. In fact, the set of derived rules can extend the set of basic rules. Derived rules take into account the context of users, namely by identifying frequently used basic rules for each user.

- **Methodology to generate intra-social basic rules:** we propose to generate a set of basic rules, within the same social network, using a photo-based methodology that utilizes the wisdom of web users. Within this methodology, we first propose a way to automatically construct the photo dataset where photos are retrieved from a content-sharing social site. In addition, we provide a set of rules based on our common sense that are independent of the relationship type to discover.

- **Methodology to generate inter-social basic rules:** we propose to generate basic rules, across different social networks, based on the characteristics of the studied social network sites. Our methodology generates basic rules by considering all profiles' attributes and features with respective weights.

- **Prototype and datasets:** we developed a prototype to validate the relevance of our rule-based methodologies and the efficiency of our relationship discovery approach. We also developed a set of tools enabling to collect real-world datasets (FOAF profiles and photos), and to create synthetic datasets (FOAF profiles). These datasets are of valuable importance for studying, through comparative analysis, social network users' interactions, and also for extracting rules for different research purposes.

1.3 Report Organization

This report is organized as follows.

Chapter 2 gives a brief overview of the basic concepts of social networks and a background review related to study social networks, namely using social network analysis and link mining

tasks.

Chapter 3 reviews related works that focus on two link mining tasks that are relevant to the realm of our work, namely 1) "Entity Resolution" and 2) "Link type prediction". We review different approaches that are related to these two tasks.

Chapter 4 presents our framework with its different components, describes its related data model by listing all the available concepts that are associated with corresponding functions to extract information.

Chapter 5 details the set of intra-social and inter-social basic rules. We describe our methodology to generate automatically each set of basic rules. We show the most important set of intra-social basic rules as well as the set of the common sense rules that we designed. In addition, we explain how to generate the set of derived rules which are obtained using mining techniques [12]. We also detail our relationship discovery algorithm with its optimization technique.

Chapter 6 presents our prototype and the set of experiments conducted to validate our approach. We describe the real-world and synthetic datasets that we use. We also detail the set of experiments to evaluate the relevance of our work for social and co-referent relationship discovery.

Chapter 7 concludes our work and presents some of our future research directions.

Chapter 2

Background

ABSTRACT

This chapter provides a brief overview of the main concepts that are essential to the understanding of this work. In the following sections, we define social networks and their common representations. This is followed by a description of the different types of data available on online social networks. We then present social network analysis, a particularly important research area to study networks. We describe its most commonly used measures and illustrate them with examples. For social network tasks that emphasis on links between social network actors, there exists a particular data mining field, called link mining. In this chapter, we detail link mining tasks, highlight some of their characteristics, and point out some examples of their applications.

2.1 Introduction

With the availability of computers and communication networks, the increase of computational power and performance, and the higher capacity to gather and analyze data, large-scale social networks studies are flourishing, spilling over all traditional disciplinary boundaries for social networks. Understanding the characteristics of these social networks, namely information related to structure, is of considerable importance and hence is presently attracting widespread interest.

At the same time, the widespread interest in analyzing real-world social networks and the explosion of the amount of social networks data have led to an increasing need to identify relevant information from hidden and implicit data. On social networks, links often exhibit rich patterns and thus can play an important role in network analysis. Links connect network actors and can be used not only to discover prominent actors within a network, but also to reveal uncovered information such as the rank of the actor, the type of a link between two actors, the category to which an actor belongs, etc.

To analyze social networks, two main categories have emerged: social network analysis (SNA) and data mining-based techniques commonly named link mining. SNA and link mining represent distinct modalities to study, analyze, and extract information from social networks.

- **SNA:** was developed by sociologists to discover the properties of social networks by focusing on social relationships between actors in a network. Several studies have been conducted since the 1930s [13] [14], exploiting global link networks' structures, identifying actors' roles and positions, and examining the interaction patterns within social networks. Although social network analysis techniques reveal important information about social networks, their main focus is to measure structural properties of networks by studying networks' topologies.

- **Link mining:** was introduced by computer scientists to extract hidden patterns from available data. Link mining can be seen as the task of applying data mining techniques on networks, while explicitly considering and emphasizing on links between social network actors [15]. The aim of data mining is to find unknown, hidden, and potentially useful knowledge from a large amount of data. In fact, data mining techniques have become vital for discovering hidden social networks' information [16].

Yet, both modalities are used to investigate social networks' characteristics although they employ different means. These techniques differ in the relative importance that they assign to each element part of social networks. While centrality measures are widely used in SNA [13],

link mining techniques rely on recent advances in data mining and often put emphasis on the links between social network actors [15] .

In the next section, we will begin our review with a brief introduction about social networks.

2.2 Social Networks

A social network is a structure comprising a set of actors that are involved together with some kinds of interactions. An *actor* is a social entity that might be a single person, a group, or a company. Actors are connected to each other through *links* that can denote one or multiple relationships. These links can be of various types, including friendship links, collaboration links, business links, etc. Consequently, one can distinguish between:

- *Heterogeneous networks:* are social networks where multiple types of actors or multiple types of links may exist (e.g., social network actors connected with different types of links, i.e., colleagues and friends).

- *Homogeneous networks:* are social networks that model a single type of actor with only a single type of link between actors (e.g., social network actors connected using only friendship links).

In practice, social networks offer to web users new interesting means and ways to connect, communicate, and share information with other members within their platforms. In theory, these social networks are made of several components, can hold different types of data, and have various representations.

Social scientists adopted two main forms to represent social networks, one based on graphs and the other on matrices. One reason behind the use of these graphical and mathematical techniques is their ability to represent and to visualize compactly and systematically the descriptions of different types of networks. While matrices are efficient for small social networks, graphs are usually used to represent networks in different fields such as computer science, sociology, biology, etc. [14] [17].

To represent social networks using matrices, rows and columns denote social actors and numbers or symbols in cells denote existing relationships, whereas graphs consist of nodes to represent actors, and edges to represent relationships. The terms *nodes* and *objects* are usually used to denote *actors*. Likewise, *edges* may also be called *links*, *ties*, or *relations*. An example

of a graph[1] and a matrix representation are illustrated in Figure 2.1(a).

In the following, we give a brief overview of graphs used to represent social networks.

2.2.1 Graph Representation

Graphs are the most straightforward way to represent actors with links between them on social networks. Sociologists refer to these graphs as "sociograms". More formally, a graph, *G = (V, E)*, consists of a set of nodes, *V*, and a set of edges, *E*. The numbers of elements in V and E are respectively denoted as $n = \|V\|$, the number of nodes, and $m = \|E\|$, the number of edges. The *ith* node, v_i, is usually referred to by its order i in the set V. A subgraph *G' = (V', E')* of *G = (V, E)* is a graph such that $V' \subseteq V$ and $E' \subseteq E$.

A graph is directed when its edges have a direction, otherwise it is an undirected graph. In an undirected graph, we refer to each link by a couple of nodes i and j such as *e(i, j)* or e_{ij}, i and j are the end-nodes of the link. Social network sites can be modeled as undirected graphs when relationships between actors are mutual (e.g., symmetric relationships on Facebook[2] where e_{ij} or e_{ji} both denote a *friendship* link between user i and user j). A directed graph is defined by a set of nodes and a set of directed edges. The order of the two nodes is important: e_{ij} denotes the link from i to j, and $e_{ij} \neq e_{ji}$. Social network sites can be modeled as directed graphs when relationships are not bidirectional (e.g., asymmetric relationships on twitter[3] where e_{ij} stands for user i is *following* user j). Figure 2.1(b) and Figure 2.1(c) show respectively a representation of an undirected and a directed graph, both with n = 5 and m = 6.

Links between social network actors play an important role in various situations on social networks. These links can be weighted or labeled within directed or undirected graphs:

- **Weighted:** weights represent the importance of relationships between social network actors. When graphs are weighted, this means that their edges are assigned with a numerical weight w that can have various indications such as link capacity, link strength, level of interaction, or similarity between the connected nodes (e.g., the number of messages that actors have exchanged, the number of common friends, etc.). Weighted graphs are used in many real-world networks. For example, they are used to measure the level of interaction in biological networks among genes, proteins and other molecules [18] [19], in bibliographic networks to measure the level of collaboration between scientists [20] [21], and in tech-

[1] In general, the terms used to denote nodes and edges vary with the social network domain.
[2] http://www.facebook.com
[3] http://www.twitter.com

nological networks to uncover important and complementary information such as in the analysis of the world-wide airport network as a weighted graph [22]. Figure 2.1(d) shows a weighted graph (on a scale of 0 to 10) where the numeric values are assigned to links and indicate the level of interaction between social network's actors.

- **Labeled:** labels are important since they can identify the type of relationships between social network actors (such as colleagues, relatives, friends, etc.). When graphs are labeled, this means that a label l is used to indicate the type of link that characterizes the relationship between the connected nodes. Labeled graphs are a popular way to model different types of structured data in various application domains. For instance, in biological networks to facilitate the study of biological resources represented as labeled graphs [23], in bibliographic networks to represent relationships that link authors, papers and publication venues [24], and in consumers/producers networks to model the roles of actors in data management relationships [25]. Figure 2.1(e) shows a labeled graph where the relationship type between linked actors is indicated.

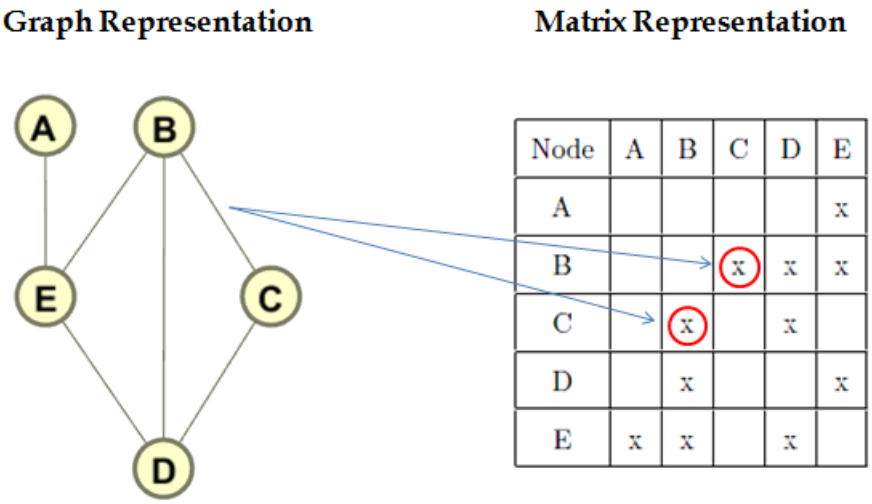
On social networks, actors create social network profiles and interact together by exchanging messages and publishing photos, as these networks allow to model the interactions among connected actors. In the following, we present different categories of data available on social networks.

2.2.2 Data

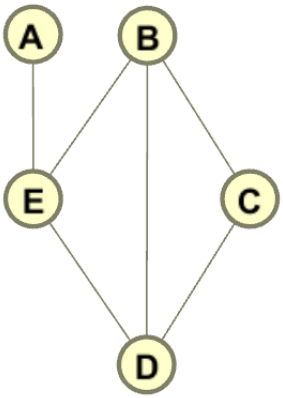
Nowadays, huge amount of information is available on the web [26] (personal data, photos, emails, etc.) from various sources of information such as email archives [27], hyperlinks implicit social structure [28], web citation information [29], biological networks [30], etc.

Traditionally, questionnaires and interviews were used to collect data from participants in order to built data networks. These conventional methods have some limitations that make them less attractive from different points of view such as:

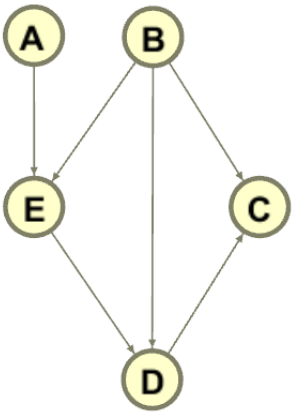
- **Scalability:** small data scale of collected information
- **Subjectivity:** participants responses can be subjectively interpreted
- **Inconsistency:** tedious process for building data network and for cleaning inconsistent information
- **Error handling:** uncontrolled errors are commonly identified.



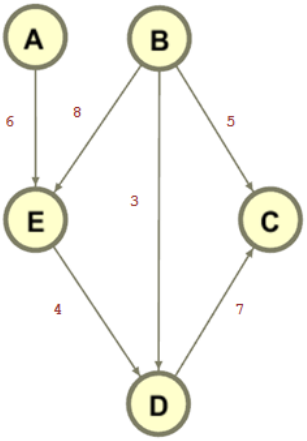
(a) Graph and Matrix representation of a social network



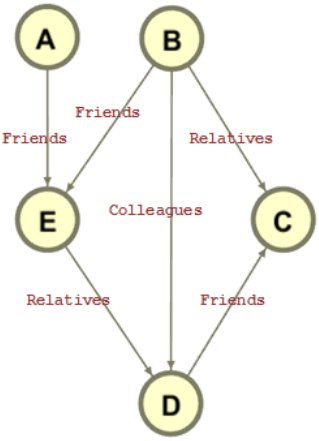
(b) Undirected Graph



(c) Directed Graph



(d) Weighted Directed Graph



(e) Labeled Directed Graph

Figure 2.1: A social network representation using a graph and its related matrix (a), an undirected graph (b), a directed graph (c), a weighted graph (d), and a labeled graph (e) with $n = 5$ nodes and $m = 6$ links.

Today, the picture has changed. As online social networks are rising around the globe, it is increasingly interesting to analyze exchanged data available over online social networks. While it is currently possible to gather data from different sources and fields in relatively short time span and often in enormous amounts, the characteristics of data differ among different social networks. In fact, the sorts of things that would be suitable for one type of data might be completely unworkable and inappropriate for another. Such data can be grouped into different categories [31]:

1. **Service data:** is the set of data the user provides to social networks to create her account such as the user's *name*, *date of birth*, *country*, etc.
2. **Disclosed data:** is what the user posts on her social network profile. This might include *comments*, *posted photos*, *posted entries*, *captions*, *shared links*, etc.
3. **Entrusted data:** is what the user posts on other users' profiles. This might include *comments*, *captions*, *shared links*, etc.
4. **Incidental data:** is what other social network users post about the user. It might include *posted photos*, *comments*, *notes*, etc.
5. **Behavioral data:** is what the social network can infer and know about the user by tracking and analyzing her activities on the site. It can be used to infer information related to the user's interests, to recommend new contacts, to display ascertainments based on the user's profile information, etc.
6. **Derived data:** is data about the user that can be inferred from all other data. For example, if 70% of your contacts live in Dijon you probably live there as well.

By analyzing social networks data, it is possible to obtain further knowledge which can be useful and precious. The availability of such data facilitates the use of different measures to analyze social networks.

In the next section, we present different measures that have been proposed in the domain of social networks to analyze the knowledge available within social networks.

2.3 Social Networks Analysis

The focus of SNA is to study social relationships between actors rather than only using their attributes. Thus, the availability of social networks data, actors, and links used to model social

networks is fundamental for analyzing these networks [14]. Recently, SNA has been used in different application domains such as email communication networks [32], mobile networks [33], coauthorship and citation networks [34], online social networks [35], terrorist networks [36], and epidemiology networks [37].

In the following, we detail the main social network analysis measures.

2.3.1 Common Measures

There are handful answers that researchers, who study social networks, would like to get such as how highly an actor is connected within a network, who are the most influential actors in a network, and how central is an actor within a network. To capture the importance of actors within a network in different perspectives, a number of measures have been proposed in the literature [38].

A commonly accepted measure is the centrality that consists of giving an importance order to the actors of a graph by using their connectivity within the network. Several metrics have been proposed to compute the centrality of an actor within a network, such as degree, closeness, and betweenness centrality [39], along with the eigenvector centrality [40]. Here, we explain each of these metrics in details.

Degree centrality: measures how much an actor is highly connected to other actors within a network. Degree centrality is a local measure since it considers the number of direct links of an actor to other actors adjacent to it. A high degree centrality denotes the importance of an actor and gives an indication about potentially influential actors in the network. With a high degree of centrality, actors in social networks serve as hubs and as major channels of information in a network. This means that they are considered as effective ways to communicate and can reach a large number of other actors. Degree centrality is applicable in 1) policy networks where action is actively present around central actors who are highly visible to other actors [41], 2) social networks to maximize the total influence for viral marketing purposes by locating highly influential actors [42], and 3) terrorist networks to build hierarchies to detect an organizational view of a corresponding terrorist network [43].

Closeness centrality: is calculated by measuring how close an actor is to all other actors. It is also known as the median problem or the service facility location problem. Closeness centrality is the length of paths from an actor to other actors in the network. Actors with small length path are considered more important in the network than those with high length path. Finding these central actors can be beneficial when the focus is on having efficient communication solutions

within networks. Closeness centrality is applied to 1) identify communities in blogs by using centrality measures and notably the closeness measure [44], 2) locate service facilities (e.g., a shopping mall) in a way where the total distance to all customers in the region is minimal [38], and 3) solve within a little time delay the task of controlling a cell's biochemistry in biological networks where central metabolites reflect both age and importance [45].

Betweenness centrality: is a measure of the extent to which an actor lies on the paths between other actors. In contrast to degree and closeness centrality which are used for characterizing influential members, betweenness attempts to analyze the social interaction in a network. It denotes the number of times an actor needs to pass via a given actor to reach another one, and thus represents the probability that an actor is involved into any communication between two other actors. Actors with high betweenness centrality facilitate the flow of information within the network. They form critical bridges between two other actors or groups of actors. A famous example that illustrates the betweenness centrality is presented in the work of [46]. In this example a group of Italian-speaking women are employed in a factory. Only one of them speaks English so to communicate with other factory workers. The single Italian woman who speaks English is the center of communication between the group of Italian women and other English speaking persons since all the communication must go through her. Such central actors control the spread of information between groups of non-adjacent actors. Betweenness centrality can be applied to 1) find communities within an email communication network where links connecting inter-community actors have high betweenness centrality while links connecting intra-community actors have low betweenness centrality [27], 2) identify hubs that most likely may cause congestion in network traffic [47], and 3) study social networks that analyze the academic performance of high school students [48].

Hence, SNA centrality metrics are intended to identify the central actor(s) in a network. Locating the central actor is however subject to the nature of the defined objective related to the central actor to find. As shown in Figure 2.2, based on which centrality measure is used (degree, closeness, and betweenness), different central actor(s) in a network may be identified.

Eigenvector centrality: quantifies the importance of actors that are connected to other central actors. Eigenvector centrality of a given actor is the sum of the centralities to its adjacent actors. In other words, the eigenvector centrality of an actor is high when it is linked to other actors having high centrality measures. Among the previously cited centrality measures, eigenvector centrality is the only one where the importance of an actor is based on the actors to which it is connected. Eigenvector centrality is applied in 1) hyperlinked web pages networks where a variant of eigenvector centrality is used to rank pages like the Google's Pagerank algo-

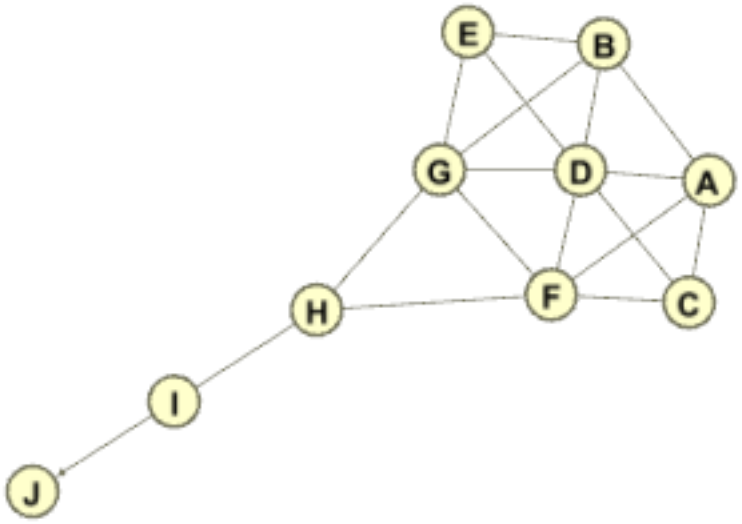
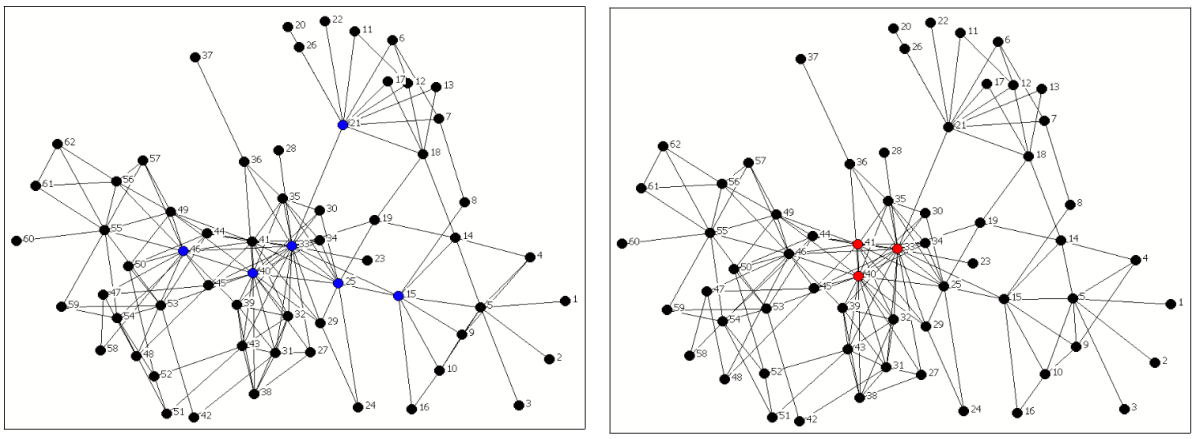


Figure 2.2: A network shaped as a kite graph where each centrality measure yields a different central actor: degree centrality (D), closeness centrality (F and G), and betweenness centrality (H).

rithm [49], 2) terrorist networks to identify the roles and the importance of actors by applying mining and using eigenvector measure on the networks structures [50], and 3) lexical networks to compute the centrality of each sentence in a cluster in order to select the most important ones to include in the summary of the cluster of sentences [51].

Figure 2.3 illustrates the terrorist network of 62 hijackers and their affiliates responsible for September 11, 2001 attacks [50]. The central actors of this terrorist network are identified using the Combine Centrality Actor Ranking measure (a measure that combines degree, closeness, and betweenness) as shown in Figure 2.3(a), and then using the eigenvector centrality measure as shown in Figure 2.3(b). To effectively break up the terrorist network, the removal of the important actors is essential to prevent future attacks. According to the objectives set by an intelligence agency, for example, the central actors to remove may not be the same depending on the centrality measure used.

The proliferation of social networks, in particular online social networks, has resulted in huge amounts of available network data. To analyze enormous volumes of data, we need to



(a) Using the Combine Centrality

(b) Using the Eigenvector Centrality

Figure 2.3: Terrorist networks represented as graphs and showing important terrorists actors (a), using the Combine Centrality (blue nodes: 15, 21, 26, 33, 40, and 46), and (b) using the Eigenvector Centrality Actor Ranking (red nodes: 33, 40, and 41).

analyze and understand social networks using adequate approaches. While studying small-world networks is straightforward (since drawing a picture of the network and answering specific questions about network structure remains possible), studying large-scaled networks remains challenging. Many findings now argue that it is possible to infer new knowledge related to the network characteristics [52]. Uncovering hidden data on large-scaled networks requires, as we believe, a more global confluence of work in different research areas [53] [54] [55] [56], a global confluence that the progress in these domains makes possible.

In the next section, we discuss how to face such challenge and review some techniques that can be used to uncover hidden information from large-scaled networks.

2.4 Link Mining

Link mining includes a set of techniques used jointly to accomplish some node-related, link-related, and graph-related tasks for the analysis of social networks. Among the most widely techniques used in link mining we can mention link analysis [53], hypertext and web mining [54], relational learning and inductive logic programming [55], and graph mining [56]. By building predictive models (predicting attribute' values) or descriptive models (extracting interesting patterns) [57], link mining can be regarded as data mining applied on social networks where links play a major role. In fact, links between actors, that might englobe more than a single

relationship, are a basic and important element in social networks.

In the following subsections, we detail all the tasks that link mining embodies as presented in [15].

2.4.1 Node-related Tasks

Node-related tasks include 4 main tasks which are:

- Link-Based node ranking
- Link-based node classification
- Link-based node clustering
- Link-based node identification.

In the following, we present each task and describe its main characteristics and domains of applications.

2.4.1.1 Link-Based Node Ranking

The objective of link-based node ranking is to prioritize corresponding actors based on their importance with respect to a chosen measure. In link mining, centrality measures (degree, closeness, betweenness, eigenvector, etc.) that exploit the network structure are used to rank the actors. There is a wide range of real-world applications that can use link-based node ranking. Table 2.1 lists various networks where link-based node ranking approaches have been used.

Table 2.1: Link-based node ranking applied on different types of networks

Domain	Aim
Bibliographic networks	indicate the impact and the rank of an author [58]
Hyperlinked web pages networks	rank pages that are connected to each other by hyperlinks in the context of the world wide web. The PageRank [59] and HITS [60] algorithms are the most well-known ranking approaches in this area
Linked documents networks	measure document similarity by analyzing the content [61]

Dealing with dynamic networks where relational data can change over time, and with large networks characterized by uncertainty is regarded as an important research direction in the context of dynamic networks [62]. One challenge in these large and dynamic networks is to maintain

updated the dynamic relational data. Another challenge is to be able to detect interesting trends like nodes whose importance may increase rapidly. Most of the current algorithms are designed for static networks [63]. However, it is interesting and challenging to develop new algorithms for node ranking within dynamic networks.

2.4.1.2 Link-based Node Classification

Among several node related tasks, the link-based node classification task classifies nodes of a network to a finite set of categories. This type of classification is not only based on nodes' attributes but also on their links to other nodes and on the attributes of these linked nodes. Link-based node classification tasks have been applied on different types of networks as presented in Table 2.2.

Table 2.2: Link-based node classification applied on different types of networks

Domain	Aim
Bibliographic networks	predict a paper's topic by considering not only the words used in the page, but also other papers that cite this paper, the citations included in the paper, etc. [64]
Biological networks	classify groups of genes, within proteins interaction networks, based on the similarity between genes profiles and on the protein products that often interact with genes [65]
Epidemiology networks	predict the type of disease based on the patient's attributes (e.g., age, weight, ethnic, etc.), and on other attributes related to the persons with whom the patient has been recently in contact [66]
Linked documents networks	predict the category to which a web page belongs based on words that occur in web pages, the links between web pages, the anchor text which is the hyperlink word on which the user clicks to access the web page, etc. [67]

A key challenge for link-based node classification algorithms is to exploit the correlation between the nodes. For instance, to be able to infer the category of a web page 1 (WP1), the category of web page 2 (WP2) must be used. Similarly, to assign a category to WP2, the category of WP1 must be known. In fact, most of the real-world data networks are mainly heterogeneous datasets with correlated linked nodes [15]. To provide a solution and improve the classification results, it is crucial to design new link-based node classification algorithms that jointly classify nodes using collective classification.

2.4.1.3 Link-based Node Clustering

Node clustering, also called group detection, is another well-studied link mining task. Its objective is to identify similar nodes and cluster them together without having predefined labeled categories. Any two nodes, members of the same cluster, are more similar to each other than to any node in other cluster. They represent communities where the level of interaction or communication (emails, messages, collaborations, etc.) between actors of the same cluster is higher than with any actor in other clusters. Table 2.3 presents three types of networks on which group detection has been applied.

Table 2.3: Link-based node clustering applied on different types of networks

Domain	Aim
Bibliographic networks	put in the same groups authors who share common research interest and collaborate with each other [68]
Criminal networks	find the structure and organization of criminal networks to facilitate investigations [69]
Social networks	allow the identification of different communities where members share the same type of activities, are interested in the same hobbies, or seek the same kind of services. Furthermore, link-based node clustering approaches enable to create clusters for persons who share different relationship types (family members, classmates, co-workers, etc.) [70]

Typically, clustering similar nodes together takes into account both the attributes of nodes and the links among them. Missing attributes, noisy data, dynamic networks can make this task more complex. It is therefore necessary to overcome these issues when clustering nodes. However, it is infeasible to infer the presence of implicit groups by analyzing whole networks since enormous amount of data is currently available on social networks. Consequently, proposing new approaches that scale up node clustering to large networks is an interesting research topic.

2.4.1.4 Link-based Node Identification

Link-based node identification, or entity resolution, is a topic that has received a lot of attention in the literature. It aims to identify nodes in datasets that have different identifiers while referring to the same real-world entity. In this case, these nodes form a matched entity pair. Entity resolution has been studied under different names such as record linkage [71], duplicate detection [72], merge/purge [73], reference reconciliation [74], object identification [75], etc. This topic has been studied in different networks as shown in Table 2.4.

Table 2.4: Link-based node identification applied on different types of networks

Domain	Aim
Biological networks	integrate information from biological databases to easily enable cross-database queries [76]
Database management networks	reconcile and integrate references from various sources that correspond to the same real-world entity [74] [77]
Lexical networks	determine which noun phrases refer to the same underlying entity within a text [78]
Social networks	determine whether two or more social network profiles refer to the same or different real-world persons [79] [80]

Entity resolution is associated with many challenges: 1) it is computationally expensive since comparing all pairs of items is not always feasible, 2) its accuracy depends on the available attributes to use, and 3) it depends on the typographical errors, abbreviations, and inverted words order which can make the task more complex.

2.4.2 Link-related Tasks

Link-related tasks are numerous and can be categorized into two different groups:

- Link prediction
- Link type prediction.

In the following, we detail these two tasks and list some networks on which link-related tasks have been applied.

2.4.2.1 Link Prediction

Link prediction, or link existence prediction, is the task of inferring the existence of a link between two nodes, based on the properties of the nodes. Link prediction is illustrated in Figure 2.4. While link prediction in static networks aims at inferring missing links and facilitating the task of creating links, link prediction in dynamic networks consists of predicting the snapshot of links at a future time. Table 2.5 lists some networks where link prediction approaches have been applied.

In the work presented in [86], the authors identified four problems related to link prediction:

Table 2.5: Link prediction applied on different types of networks

Domain	Aim
Bibliographic networks	infer the existence of a link from indirect evidence such as co-authorship of a paper or a large number of co-authored papers [81] [82]
Biological networks	predict missing protein-protein links by using the topology of an observed protein interaction network [83]
Digital libraries	analyze user-item interactions for making collaborative filtering recommendations [84]
Social networks	predict the existence of friendship and family links in social networks using both of the attributes and the structural features [85]

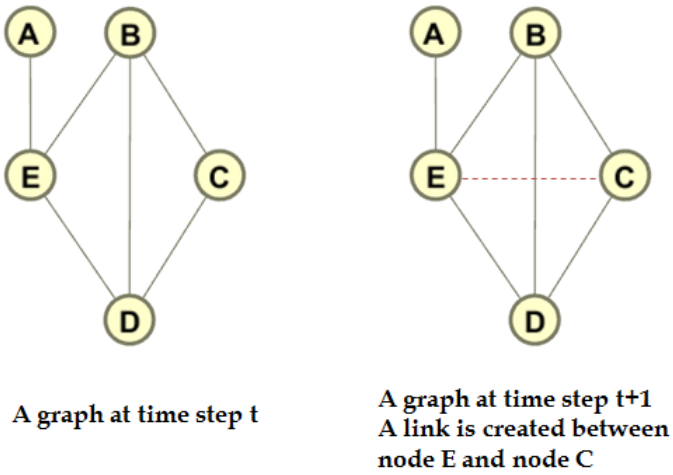


Figure 2.4: Example of link creation in a graph between time step t and t+1.

- **Random graph models:** consist of defining models to generate graphs that capture the properties of graphs in real networks [87]
- **Link completion:** tries to propose the best nodes for an action (e.g., in an email communication network, link completion tries to find the best set of recipients for a message) [88]
- **Leak detection:** consists of inferring which nodes shouldn't be part of an action (e.g., recipients who must not receive a message in the email communication network) [89]
- **Anomalous link discovery:** identifies links that are statistically improbable so to improve the analysis of a network [90].

These listed problems pose significant challenges for link prediction in terms of meeting applications' objectives, and therefore it is important that new approaches provide possibilities to deal with any confusion (anomalous link discovery), network misunderstanding (random graph models), and noise (link completion and leak detection) .

2.4.2.2 Link Type Prediction

Unlike link prediction, where the aim is to predict the existence of a link between two nodes at a particular time, link type prediction tries to identify the type of an existing link. Here, it is assumed that we know that a link already exists between the two nodes. Table 2.6 shows some examples where link type prediction methods have been applied.

Table 2.6: Link type prediction applied on different types of networks

Domain	Aim
Bibliographic domain	identify the type of the link between two co-authors (advisor-advisee, co-authors, etc.) [81]
Hyperlinked web pages networks	identify whether the type of a link is an advertising link or a navigational link [91]
Social networks	identify the type of links connecting users to each other (friends, relatives, colleagues, etc.) [92] [93]

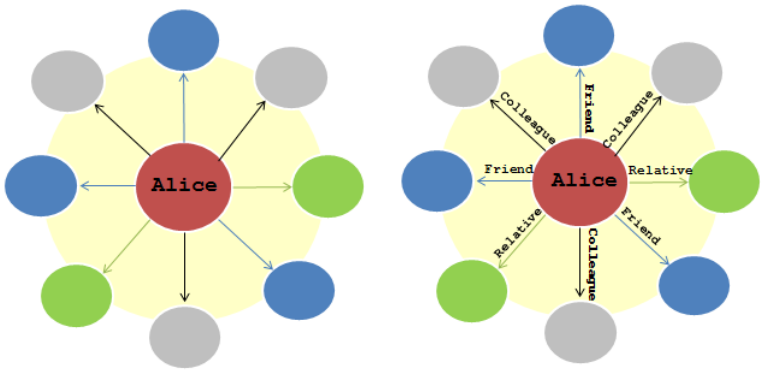
Figure 2.5 illustrates two social graphs of a main user where the left graph has no link type between the main user and her contacts, meanwhile the second graph on the right has its links labeled with the type of relationship between the main user and her contacts.

Applying link type prediction is of great importance to avoid the daunting task of manually labeling the links. In addition, knowing links' types could be critical toward better node classification for many tasks, and crucial for various privacy applications.

2.4.3 Graph-related Tasks

We can distinguish between three main groups of graph-related tasks:

- Subgraph discovery
- Graph classification
- Graph-based generative models.



(a) Unlabeled Graph (b) Labeled Graph

Figure 2.5: Example of two graphs of a main user where in the first the links with the user's contacts are unlabeled (a), and in the second links with the user's contacts are labeled with the corresponding identified link type.

In the following, we present and describe each of these three graph-related tasks.

2.4.3.1 Subgraph Discovery

Subgraph discovery is a link mining task that attempts to detect similar substructures in pairs of graphs. It is one of well-studied topics related to graph mining where graphs are natural representations of diverse complex structures. Frequent subgraph discovery approaches are successfully used on graphs where each actor's label is used only once per graph [94]. This type of graphs is called *relational graphs*. Subgraph discovery approaches have been applied on different networks as shown in Table 2.7.

Table 2.7: Subgraph discovery applied on different types of networks

Domain	Aim
Biological networks	discover motif of repeated structures in the network for protein interactions that may be of biological interest [95]
Software behavior networks	discover signatures that are bug relevant by extracting the most discriminative subgraphs of correct and faulty runs [96]
Terrorist networks	recognize automatically terrorist-related web sites in multiple languages [97]

Two key emerging challenges for subgraph discovery address the following problems [98]: 1) the size of networks where an enormous amount of data is to be processed: dealing with the whole set of links and nodes is computationally expensive, and 2) networks that change dynamically

over time, also known as streaming networks: as new links are incrementally received, subgraph discovery task is supposed to process the received data in real time.

2.4.3.2 Graph classification

Graph classification aims at classifying an entire graph with respect to a specific category. Table 2.8 lists some networks where graph classification approaches have been used.

Table 2.8: Graph classification applied on different types of networks

Domain	Aim
Biological networks	classify protein networks where a protein is assumed to perform the same function as the most similar protein in a database of known proteins [99]
Multi-agent networks	decide when it is the best to join or to leave a multi-agent network. This is highly indicative for profitable participation in supply chain networks [100]
Software behavior networks	classify a set of incorrect and correct executions represented as graphs based on observations of program behaviors [101]

Classifying each actor in a large graph is a tedious and sometimes infeasible task. Rather than trying to label each node within a graph, graph classification categorizes the whole graph by applying labels to the entire graph. However, it is important to note that approaches based on graph mining algorithms [102] [103] or graph kernels similarity [104] [105] fail to scale up with large networks. In fact, they become prohibitively inefficient and expensively costly in terms of processing [106]. Consequently, it is inevitable that more scalable graph classification algorithms are needed.

2.4.3.3 Graph-based generative models

Generative models for graphs try to understand the characteristics of networks, to identify the mechanisms of networks growth and evolution, and to generate networks with realistic properties given few parameters. Generative models have been applied on different networks as shown in Table 2.9.

Studying generative models for graphs is becoming increasingly important, in particular when temporal metrics are considered i.e., the network changes over time. In particular, one of the main characteristics of social network is that social relationships evolve between users,

Table 2.9: Graph-based generative models applied on different types of networks

Domain	Aim
Biological networks	predict links using network structures where generative processes are used to model the network with latent features in a protein interaction network [107]
Communication networks	study how relationships evolve between actors in order to detect new communities by measuring the communication frequencies and considering the semantic of information [108]
Email networks	discover latent communities using both available links and content information from an email communication network [109]

which means that new links can appear and others can disappear. Therefore, it is interesting to apply generative methods to understand and to reveal the future trend of network evolution.

Summary

Since different link mining tasks can be applied on the same network for various aims, it is interesting to identify which networks are used by which link mining task(s). Within the networks we mentioned in this chapter, we assign each network to its corresponding link mining task(s) as presented in Table 2.10. Networks which have been used for:

1. **One link mining task:**
   (a) Node-related tasks: criminal, epidemiology, and linked document networks
   (b) Links-related tasks: database management, digital libraries, and lexical networks
   (c) Graphs-related tasks: multi-agent, software behavior, and terrorist networks
2. **Two link mining tasks:**
   (a) Node and link-related tasks: bibliographic networks
   (b) Node and graph-related tasks: hyperlinked web pages networks
3. **Three link mining tasks:** node, link, and graph-related tasks have been applied on biological and social networks.

Table 2.10: A summary of link mining tasks applied on different types of networks

Network type	**Node-related tasks**	**Link-related tasks**	**Graph-related tasks**
Criminal	✓		
Epidemiology	✓		
Linked document	✓		
Database management		✓	
Digital libraries		✓	
Lexical		✓	
Multi-agent			✓
Software behavior			✓
Terrorist			✓
Bibliographic	✓	✓	
Hyperlinked web pages	✓		✓
Biological	✓	✓	✓
Social	✓	✓	✓

2.5 Conclusion

Understanding the characteristics of social networks, or more generally networks, has been among the most studied topics in the social network analysis community [13]. Two main research areas have focused on studying social networks: 1) social network analysis, and 2) link mining. Their application domains range over different types of networks, among which we cite biological networks, bibliographic network, epidemiology networks, terrorist networks, social site networks, etc. In this chapter, we provided a review of a number of predictive and descriptive models that have been built to address different tasks. Generally, each task corresponds to a defined real-world challenge to solve. In this work, we set our focus on two social network related tasks: the first, "Entity Resolution", a node-related task , and the second, "Link Type Prediction", a link-related task.

In the next chapter, we provide a review of approaches related to these two tasks.

Chapter 3

Related Works

ABSTRACT

The focus of this work is on understanding and analyzing interactions arising within a single or across multiple social networks in order to semantically enrich links that connect users on social networks. Link mining has been among the most widely studied topics that emphasize on the links for the analysis of social networks. Among several link mining tasks, we are mainly interested in two tasks: (1) "Entity Resolution" and (2) "Link type prediction". The following literature review is divided into two main sections. We first present related works in the area of "Entity Resolution". We cover general and social network-oriented entity resolution approaches. Then, we provide a review on different approaches related to "Link type prediction" on social networks. In this chapter, we summarize and discuss the limitations and features of the outlined link mining approaches.

3.1 Introduction

For the past few years, there has been a rapid rise in the use of social networks. The ever-growing number of social network users and the huge amount of published information have reflected the need to bring more semantic features to users and applications. With this increasing volume of published data, different problems related to social networks abound, from contact management [8] to data management [110], and from privacy protection [111] to reputation management [112].

One of the most intriguing social network challenges addresses the problem of contact and relationship management. Indeed, users may get connected to different types of contacts. This diversity, yet the different levels of social closeness between users and their contacts, entailes an increasing need to analyze social interactions for better user-profile management. Currently, users are often provided with a link connecting them to each of their contacts within a single social network site. However, as users can be connected with various types of contacts (colleagues, relatives, friends, etc.), links with labeled relationship types are needed but are rarely explicitly represented on social networks.

At the same time, social networks are attracting a widespread interest thanks to the particular services and functionalities that each site offers to target a well-defined community in the real world. To make use of the provided services/functionalities and to stay tuned with its related members, users create several accounts on various sites where they disclose personal and professional information. Social network users can be connected to their contacts either on one or several sites. Multiple, yet different user profile representations, may refer to the same real-world person. As a result, users can be connected to contacts who may have duplicated profiles across different social networks.

Beyond the scope of each social site, available data on social networks lack semantic and interoperability. Hence, when exploiting user profiles and their links' characteristics, two main challenges arise:

1. **Within a single social network:** where the type of the link between social network users and their contacts is missing. Effective solutions to enrich links with types stating whether two persons are colleagues, relatives, or friends, etc., are yet missing.

   With the growing success of online social networks, there is an increasing need to discover social relationships among users. This is relevant in several scenarios such as identifying persons using face annotation techniques in personal photo collections [113], implementing marketing strategies over social networks [114], protecting users privacy [115] [116], and

improving face clustering tasks [117] [92].

2. **Across multiple social networks:** where duplicated contacts are not identified. It is important to provide solutions to enrich links with properties stating whether two persons refer to the same real-world person or two different real-world persons.

   Entity resolution can be regarded as the task of creating virtual links between contacts who refer to the same real-world persons. Different scenarios could highly benefit from semantic features embedded in these links [118] [119] [120]. This is relevant in data cleaning scenarios [118] for meaningful data analysis and in data integration scenarios to correctly combining unique representations of real-world entities [119] [120]. A noteworthy benefit in social networks is that entity resolution allows to detect co-referent users in order to combine and to merge their profiles into a single, global, and more complete profile representation [121].

Effectively exploiting the available data on social networks could greatly benefit social network users and different applications by turning hidden information into explicit and ready to use data. In fact, the knowledge extracted from users' profiles and their interactions can be used to identify different links' characteristics with the help of semantics. Building on top of link mining techniques and social network data, semantically enriching links is not trivial. Such task involves, in one way or another, the discovery of the types of links and the matching of users' profiles. Link type prediction and entity resolution are wide and active research areas, both part of link mining tasks. We believe that they are tightly correlated and investigating them together is a novel and interesting topic.

In the next section, we provide a review of previous studies that have addressed the topics of entity resolution and link type prediction.

3.2 Entity Resolution

Entity Resolution comes in many guises and for many different applications, varying from database management [74] [77] and natural language processing [78] to biological networks [76] and social networks [79] [80]. Addressing the entity resolution problem presents many challenges [122]. Among them, the absence of a global unique identifier across different social networks is the most preeminent one. In the following, we present different approaches that have been presented to reliably and uniquely identify real-world entities.

3.2.1 General Approaches

In the literature, record linkage [123] is also used to refer to the problem of entity resolution. Formalized under a probabilistic model by the authors of [71], different record linkage approaches have been developed to compare the textual similarity among the entities' attributes [123] [71] [73]. More recently, other approaches that don't exclusively depend on a single attribute have been proposed [124] [77] [125] [74] [126] [127]. These approaches suggest, in addition to comparing main entities' attributes, to consider the comparison of the information of entities that are linked to the main entities. Approaches that deal with the problem of entity resolution can be categorized into three groups:

1. Attribute-based entity resolution approaches
2. Naive relational entity resolution approaches
3. Collective relational entity resolution approaches.

To illustrate these approaches, let us take four examples of two main users on two different social networks:

- **Example 1:** Same real-world main user and same real-world linked entity (c.f. Figure 3.1(a))
- **Example 2:** Same real-world main user but different real-world linked entities (c.f. Figure 3.1(b))
- **Example 3:** Different real-world main users but same real-world linked entity (c.f. Figure 3.1(c))
- **Example 4:** Different real-world main users and different real-world linked entities (c.f. Figure 3.1(d)).

In the following subsections, we present the three categories of entity resolution approaches. Then, we provide a review of approaches designed specifically to the context of social networks.

3.2.1.1 Attribute-based Approaches

Attribute-based approaches, or feature-based approaches, are based on computing the distance among a pair of attributes.

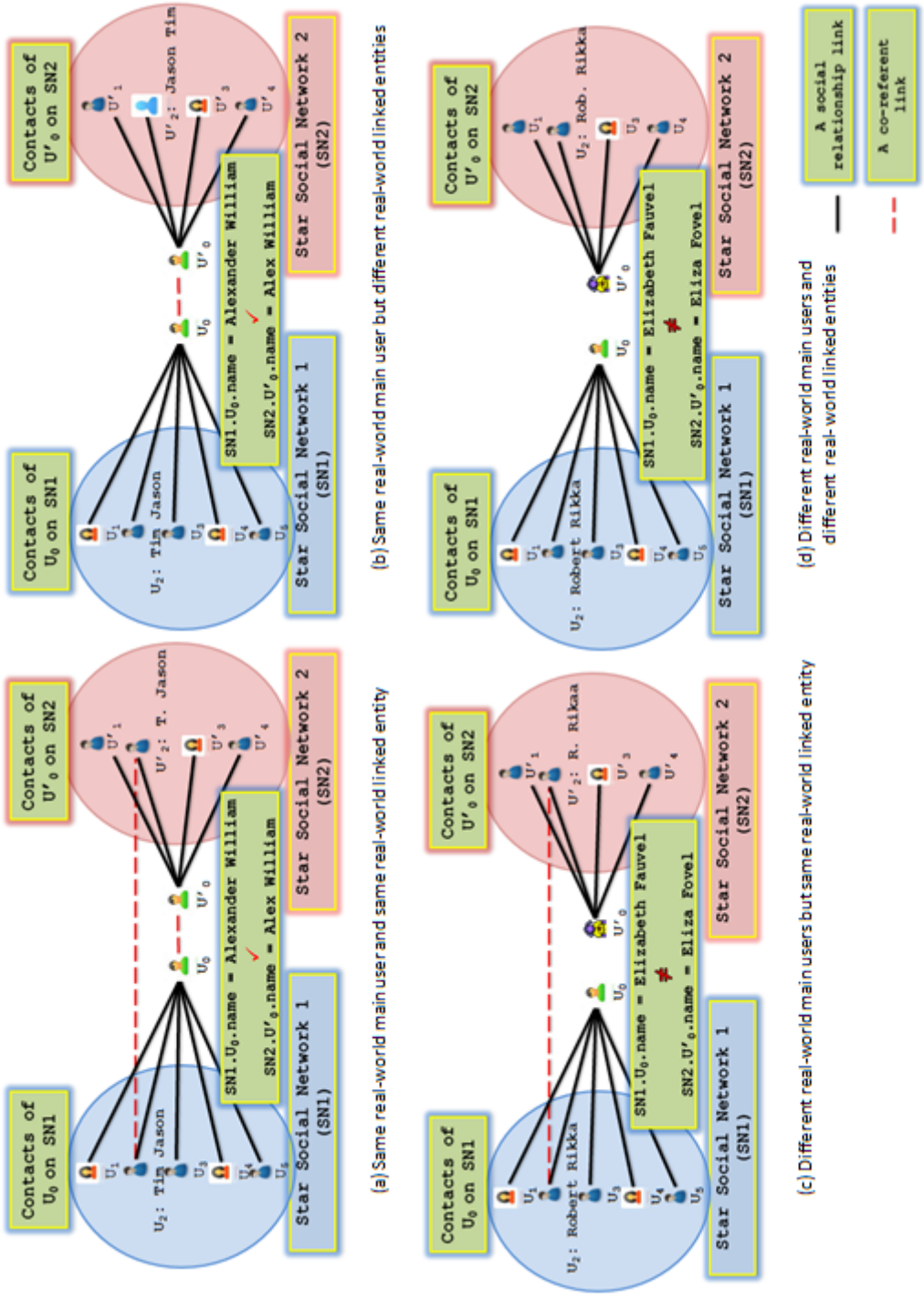


Figure 3.1: Four examples related to entity resolution.

Many approaches [128] [129] [130] have been proposed based on the work presented in [71]. If the distance (Levenshtein Distance [131], Jaro [132], Jaccard [133], etc.) between two attributes is below a defined threshold, the two corresponding entities are considered similar and thus co-referent. It is to be noted that the similarity between two attributes is inversely proportional to their distance, so the similarity is the highest when the distance is 0, and the lowest when the distance is 1. Throughout this chapter, to compute the similarity between two attributes, we consider the *name* of the person as a matching attribute, the Jaro distance as a default distance, and a manually predefined matching threshold of 0.75. For instance in *Example 1* (c.f. Figure 3.1(a)) the similarity between the name of $SN1.U_0$ "Alexander William" and the name of $SN2.U'_0$ "Alex William" is 0.82. In this case, the two persons are considered to refer to the same real-world person since their similarity is above the predefined threshold (0.75).

Unfortunately, the methods of this category face two major drawbacks:

1. Choosing an appropriate merging threshold and selecting the best matching attribute(s) are not always an easy task [134] and can be susceptible to false positive matches. Given the predefined threshold of 0.75 over the *name* attribute, two persons are considered as referring to the same real-world person only when the similarity between their names is above a predefined threshold. This is not the case in *Example 3* (c.f. Figure 3.1(c)) where $SN1.U_2$ "Robert Rikka" and $SN2.U'_2$ "R. Rikaa" have a similarity score of 0.69, even though that they refer to the same real-world person.

2. Distinguishing between the representations of two different real-world entities is impossible when these entities have the same or similar values over a predefined matching attribute. For example, given the *name* of the person as a matching attribute and the predefined similarity threshold of 0.75, two different real-world persons cannot be identified as different real-world entities, even though they are. The similarity between $SN1.U_0$ "Elizabeth Fauvel" and $SN2.U'_0$ "Eliza Fovel" is 0.77 (*Example 4* c.f. Figure 3.1(d)).

Consequently, using only attributes is inappropriate for many real-world data where missing attributes, attributes with erroneous values, and attributes with values changing over time are frequently encountered. To effectively solve the entity resolution problem, using only attributes is not enough, and hence more reliable approaches are needed.

3.2.1.2 Naive Relational Entity Resolution Approaches

Naive relational approaches go beyond the comparison of entities based on their attributes exclusively [124] [77] [125]. Instead, naive relational approaches aim to consider attributes of the linked entities as well, requiring to incorporate and analyze the links between entities.

Several naive relational methods have been developed [124] [77] [125]. In [124], the authors propose a relational de-duplication method using edges in data warehouse applications where there is a dimensional hierarchy (e.g., city-state-country). This hierarchy is used as extra information to identify duplicated records. In [77], the authors describe an iterative de-duplication process in the presence of relations. For the task of entity resolution, their approach combines the use of attributes similarity measures and the similarity of linked entities. In [125], the authors introduce the Relationship-based Data Cleaning (RelDC) approach that exploits the relationships among entities for the purpose of disambiguation. RelDC relies on author affiliation and co-authorship information to identify co-referent authors in a citation database. RelDC analyzes, among entities in the graph, the relationships by comparing the strength of their connections.

Naive relational approaches assume that the data representation is similar to the relational database. In this case, entity resolution takes into consideration the linked entities but it is limited to their attribute level. For instance, two main users $SN1.U_0$ "Alexander William" and $SN2.U'_0$ "Alex William" can be considered as the same real-world entities if their linked entities refer to the same entity, for instance, $SN1.U_2$ "Tim Jason" and $SN2.U'_2$ "T. Jason" respectively (*Example 1* c.f. Figure 3.1(a)). As a matter of fact, the decision does not only depend on the attribute similarity of the main user ($SN1.U_0$ "Alexander William" and $SN2.U'_0$"Alex William") but also on the textual similarity (0.88) of the attribute *name* between their linked entities ($SN1.U_2$ "T. Jason" and $SN2.U'_2$ "Tim Jason"). However, the main drawback of these approaches is that, when two linked entities have similar or exactly the same attribute value (this happens largely in some contexts), they are considered as referring to the same entity. For instance, in *Example 4* (c.f. Figure 3.1(d)), the two linked entities $SN1.U_2$ "Robert Rikka" and $SN2.U'_2$ "Rob. Rikka" on both social networks are similar enough (with a similarity score of 0.85) to be considered as referring to the same entity, but in reality they are not. Consequently, the two main entities $SN1.U'_0$ "Elizabeth Fauvel" and $SN2.U'_0$ "Eliza Fovel" are then erroneously considered as referring to the same real-world persons. In another case, in *Example 2* (c.f. Figure 3.1(b)), $SN1.U_0$ "Alexander William" and $SN2.U'_0$ "Alex William" are considered as different real-world persons because the similarity score between their linked entities is below the predefined similarity threshold (similarity score is 0.61 between $SN1.U_2$ "Tim Jason" and

$SN2.U'_2$ "Jason Tim").

Although these approaches are more reliable than the attribute-based approaches, they fail to deal with many cases (*Example 2* c.f. Figure 3.1(b) and *Example 4* c.f. Figure 3.1(d)). Only comparing the attributes similarity between the main entities and their linked entities may output erroneous results. Naive relational approaches are not adapted to datasets where common attributes' values are frequent (e.g., networks where same names are frequent). Therefore, it is necessary to find ways to overcome such limitation as discussed in the next subsection.

3.2.1.3 Collective Relational Entity Resolution Approaches

While naive relational approaches consider the attributes similarity between linked entities, collective relational entity resolution approaches apply entity resolution on the linked entities as well. Collective relational entity resolution approaches are motivated by the intuition to exploit the rich information available in the relationships between entities.

Different collective relational approaches have been proposed in the literature [126] [74] [127]. In [126], the authors propose to perform collective relational entity resolution on graph where linked entities are resolved jointly. They propose a graph-based clustering algorithm applied on a bibliographic network. In [74], the authors propose an iteratively reference reconciliation method among interlinked entities. They model the relationship as a dependency graph where entities are iteratively resolved and then propagated within the graph-like structure, thus allowing the discovery of more matching entities. In [127], the authors propose a solution to the entity resolution problem based on Markov logic [135]. Evidence is merged to allow the flow of reasoning between different pairwise decisions over multiple entity types.

Therefore, to overcome the problem stated previously concerning *Example 2* (c.f. Figure 3.1(b)) and *Example 4* (c.f. Figure 3.1(d)), collective relational approaches resolve the linked entities as well. They take into account the related entities of the linked entities (these linked entities are not shown in Figure 3.1). For instance, in *Example 2* where $SN1.U_0$ is "Alexander William" and $SN2.U'_0$ is "Alex William", entity resolution is applied on their linked entities, $SN1.U_2$ 'Tim Jason" and $SN2.U'_2$ "Jason Tim". As a result, even if the *name* similarity score between $SN1.U_2$ "Tim Jason" and $SN2.U'_2$ "Jason Tim" is 0.61 and consequently below the predefined similarity (0.75), collective relational entity resolution allows to conclude that these persons refer to the same real-world person since the linked entities of $SN1.U_2$ and $SN2.U'_2$ are resolved as well. Similarly, in *Example 4* when applying collective relational entity resolution on the two linked entities, $SN1.U_2$ "Robert Rikka" and $SN2.U'_2$ "Rob. Rikka", one can confirm

that these two entities refer to different real-world persons regardless of their name similarity. This represents an important evidence to conclude that the main entities ($SN1.U_0$ and $SN2.U'_0$) are also different real-world persons.

Although collective relational entity resolution allows to overcome many challenges, the scope of its utility is limited. For example, this collective relational approach can erroneously imply that two main entities are different only if their linked entities are (c.f. *Example 2* Figure 3.1(b)). In addition, one of the major drawbacks to the use of collective relational approaches, is the fact that, it is required to have explicitly available relationships, otherwise there is no guarantee that efficient entity resolution can be achieved. In other words, the presence of relationships between entities is necessary in order to be used as an additional evidence to potentially improve the effectiveness of these approaches [134]. In practice, the assumption that relationship types are always available does not hold in many contexts. For instance, in bibliographic networks, authors collaborate together and co-authorship relationships must be extracted; in social networks, relationships must be discovered where different interactions can include colleagues, relatives, friends, etc. Another important limitation is that collective relational approaches are computationaly expensive [136]. Consequently, entity resolution effectiveness might be improved, but the efficiency is compromised and can hardly scale for large networks.

3.2.1.4 Partial Conclusion

So far, we presented three major lines of approaches regarding entity resolution, and described both their advantages and limitations. In Table 3.1, we summarize the different categories of entity resolution approaches with a description of their features and limitations.

Our review shows that:

- Dealing with real-world datasets with no missing attributes' values is rarely possible. Missing attributes are one of the primary limitations for entity resolution approaches.
- Attribute-based approaches suffer from mistyping errors, the use of abbreviations, inverted word order, etc. They are highly dependent on the similarity function used as well as on the defined similarity threshold.
- Entity resolution approaches need a sufficient and a valid amount of data to be available. However, it is common that two datasets don't have a sufficient number of attributes to decide if two entities represent the same real-world entity or not, e.g., a user willing to merge her contacts on a smartphone with her contacts on a mail account.

- Naive relational and collective relational approaches assume the presence of labeled relationships in order to operate. In reality, this is not always true, i.e., social networks have rarely their relationships typed.
- Most of the relationship-based approaches focus on the bibliographic and citation domains, and thus it is difficult to apply them directly in other domains [74] [126].

In the next section, we carry on our review to cover approaches that have addressed the entity resolution problem within the context of social networks.

Table 3.1: Summary of the entity resolution approaches

Approaches	Features	Limitations
Attribute-based [128] [129] [130]	- Are applied on the attributes of the main entities - Are suitable to deal with simple typographical errors and with noisy entities' attributes - Are appropriate to use in simple entity resolution scenarios (e.g., customers of a business dataset, patients of a medical dataset, or students in a school dataset, etc.)	- Limited to the main entity - Depend on the similarity threshold and on the similarity function - Suffer from complex mistyping errors, the use of abbreviations, inverted word order
Naive relational [124] [77] [125]	- Incorporate the attributes of the main entities and the linked entities into the score - Take into account the structural similarity (relationship between entities)	- Only take into consideration the attributes of the linked entity - Relationships must be explicitly present - Not adapted to datasets where common attributes' values are frequent (e.g., names)
Collective relational [126] [74] [127]	- Resolve the identity of the linked entities - Provide significant evidence to determine co-referent entities by jointly evaluating related references	- Computationally expensive - Relationships must be explicitly present

3.2.2 Social Network-oriented Approaches

In the context of social networks, "user profile matching" is the term commonly used to refer to the problem of entity resolution. Similarly to the concept of an Inverse Functional Property (IFP)

from Web Ontology Language (OWL) [137], which refers to the notion of a key in databases, FOAF[1] makes use of a set of IFPs to identify co-referent profiles. FOAF defines several attributes as IFP, such as *foaf:mbox*, *foaf:mbox_sha1sum*, and *foaf:homepage*.

In this section, we present social network-oriented approaches that have been suggested to solve the problem of user profile matching. In the first category, we start by presenting the approaches that are based on classical unique identifiers [138] [139] [140]. In the second category, we provide descriptions of approaches that propose their own unique identifiers [141] [142] [143]. Finally, in the third category, we include approaches that propose solutions that go beyond the use of unique identifiers [144] [145] [80].

3.2.2.1 Classical Unique Identifiers

These approaches assume that the two profiles, represented mainly using FOAF, refer to the same real-world person whenever they hold the same IFP value.

In the work presented in [138], the authors propose a heuristic approach to identify, discover, and analyze FOAF documents from the Web. Information is extracted from FOAF documents about different persons. The authors analyze a set of collected FOAF documents and formulate a set of observations: When a same person is described in several FOAF documents, information from these documents is combined to form aggregated information about the person. To combine this information, the authors consider that the FOAF unique identifiers such as *foaf:mbox_sha1sum*, and *foaf:homepage*, are the ideal clues. However, other identifiers such as *foaf:name* may also be useful in giving some further clues. The authors urge for caution when merging information from many FOAF documents since some of the facts may be wrong thus resulting into contradictory information.

Flink, a system detailed in [139], is used for the extraction, aggregation and visualization of online social networks. It collects information from different sources such as web pages, emails, publication archives, and FOAF profiles. Identity reasoning (or smushing) is included in the system in order to identify profiles that refer to the same persons across multiple information sources. This reasoning is based on name matching and on IFP comparison implemented within its code. For the name matching, the similarity is computed between two names (the dissimilarity between last names is not allowed). As for the used IFP, the system employs a set of properties, such as mailbox, mailbox checksum, and the homepage.

In [140], the authors present a work in which they study the state of the semantic web

[1] http://xmlns.com/foaf/spec/

by reasoning on FOAF profiles. They consider the *owl:InverseFunctionalProperty* as the most important semantic feature. Their reasoning on FOAF is based on the *foaf:mbox_sha1sum* IFP to infer whether two profiles refer to the same real-world person or not. They qualify this as a critical inference where the OWL reasoner's decision is considered always logically correct. In their study, they discover that thousands of users have accounts on multiple social networks, linking their subgraphs in the unified social network.

To sum up, the previously proposed approaches based on IFP [138] [139] [140] depend mainly on a chosen or an already defined IFP attribute in order to identify two users' profiles as co-referent. Although an IFP is designed to uniquely identify persons, its current use is far from being able to meet its principal functionality. While many attributes are misused (*foaf:weblog*, *foaf:homepage*), the use of the email address, which is the main IFP, does not guarantee a reliable solution. Although the value of the email address is not always available or hidden, the use of the email address suffers from a plethora of pitfalls [141]:

- Users change email address (change work/educational institution, choose better provider, drop over-spammed email address, etc.).
- Users have more than one email address depending on the context (work, online gaming and shopping, family and friends relationships, etc).
- Email addresses can act as proxies for more than one person.

Consequently, the used IFP attributes cannot be considered as an appropriate global identifier for the task of profile matching on current social networks. They represent only one evidence that two profiles may refer to the same real-world person.

3.2.2.2 Case-specific Unique Identifiers

These approaches propose to create and use their own global unique identifiers, different from the FOAF IFP, in order to identify co-referent users.

In [141], the authors consider that matching user profiles using an IFP, such as the *foaf:mbox_sha1sum* in FOAF, is not suitable. They provide some explanation showing that it is very common for a user to have two social network accounts with different email addresses. They present an extended service called Foaf-O-Matic for the creation of FOAF profiles. The main functionalities of this service are: uploading a FOAF profile, describing the primary person, adding and describing friends by relying on the Okkam infrastructure [146], selecting one Okkam entity for

each described person, and retrieving the new FOAF description. The Okkam infrastructure is a central repository that provides global unique identifiers for users.

In [142], the authors investigate separately two approaches that can identify the co-occurrence of the same person across different communities. The first approach is based on the IFP. The second one is based on heuristics, in particular label-based similarity using a simple but strict string comparison technique. The result of both approaches reveals that there exists persons who have more than one e-mail account and there are different persons with the same name. In general, the profiles that refer to the same real-world persons based on their IFP are largely included in the label-based results.

Some works, such as in [143] also defined their own IFP attribute that suits their needs. Here, the authors seek to achieve instance integration by merging instances of the same papers coming from two computer science bibliography databases: CiteSeer [147] and DBLP [148]. Therefore, they propose to uniquely identify publications by defining the electronic edition URI property of a paper called *pub:ee* as an IFP. This allows to achieve instance integration over datasets by considering two publications as equivalent if they have the same value for the *pub:ee* property.

Generally speaking, these approaches present three main drawbacks. First, in [141] [143] users must add and identify manually each friend as well as determine by themselves duplicated contacts' profiles. Second, such approaches have their scope limited since a significant number of existing profiles, or entities in general, may not include their defined IFP. Third, only using IFP or only using information retrieval technique, as presented in [142], would limit the capacity of the system in order to obtain good results. Extending current profile representations vocabularies, or proposing new standards, where a required global IFP exists is a complex and infeasible task. Therefore, it is inevitable to propose solutions that are well-adapted to the current state of social networks that make use of available technologies and different research fields (information retrieval, data mining, decision making, etc.).

3.2.2.3 Beyond Unique Identifiers

To match user profiles, several approaches suggest to extend the use of IFP by proposing new methods that don't strictly depend on the IFP values.

In the work presented in [144], the authors propose to disambiguate the identity of users by using social circles. Social circles represent a group of persons linked to a central user by some identifiable common relations. To do so, they extract data related to social network users from social circles. Such data can include different identity features like blogs, images tagged with

person names, and instant messaging conversations. It is then up to the users to decide which identity features are best suited to minimally distinguish their identity from others.

In [145], the authors apply logic and numeric entity reconciliation between two social sites to identify collaborative friend relationship. The FOAF vocabulary is used as a mediation layer to cover the heterogeneity between different data sources by mapping their semantic to the new extended schema. In their work, they extend FOAF by adding new relationship types to the existing ones *foaf:knows* (*clique co-author*, *common co-author*, and *co-author*). For the logic reconciliation, the authors use the so-called approach L2R [149], a Logical method for Reference Reconciliation, that creates firstly rules to match different vocabularies between two networks. Next, they apply a numerical method to match values. The authors define this numerical method as the Probabilistic Semantic Model (PSM) that extends standard attribute-based Bayesian network to resolve atomic and associated properties of semantic models. For the numerical reconciliation, semantic similarity is used to identify similar identities based on their properties values.

In the work presented in [80], the authors propose a methodology to produce a classifier for identifying co-referent FOAF profiles. They consider this problem as a classification task and use the Support Vector Machine (SVM) [150]. In this work, only a subset of the FOAF attributes is used by the authors. The selection of attributes is based on the likelihood that the selected subset is able to output the best classified data. They built a training and a test dataset made of profiles downloaded from both blog and non-blog web sites. Profiles that are unlikely to match and hence useless for constructing the training set are ignored. The remaining profiles are processed by measuring the string similarity between their common properties. Different measures are described between profiles' attributes: 1) an exact string match for each property, 2) a partial string match, and 3) a cross string match between different attributes (e.g., name and nickname). When the similarity score is above a manually defined threshold, the corresponding pair of profiles is kept for the training. As for the test dataset, the authors construct it by using Yahoo social networking site[2] from where they download profiles of users who have indicated that they have a twitter[3] account. They used a FOAF generator based on twitter accounts to get a second FOAF profile for each user. Using this training set, the classification of new pairs of profiles from the test set can start.

To conclude, social network-oriented approaches that go beyond the use of IFP [144] [145] [80] are interesting but they suffer from some drawbacks. In fact, letting the users choose the attributes, as in [144], that they consider best suited to distinguish their identity from others, can

[2]http://www.mybloglog.com
[3]http://www.twitter.com

be a difficult task for them and may output uncertain results for profile matching. Furthermore, in the work of [145], the decision of matching profiles is based on the name, the first name, the email address, and on whether the friends within the profiles refer or not to the same real-world persons. These attributes may not be enough or may be inappropriate for the comparison of other social networks. This method is not convenient if the semantic structure of the social networks is more complex. Similarly, in the work of [80] a subset of attributes is used and a similarity threshold is defined manually. The training set doesn't take into consideration the context of each user, and fails to automatically extend its knowledge. Like all the previous methods, the methods in this category are limited to textual attributes.

3.2.2.4 Partial Conclusion

Obviously, the entity resolution task on social networks is more complex than just using unique identifiers. With the current state of social networks that are functioning as "Data Isolated Islands" [151], the FOAF IFP attributes cannot be considered as global identifiers for the task of profile matching. At the same time, more flexible unique identifiers that allow to identify co-referent users would be useful. However, it is clear that manually defining unique identifiers is not the solution. From this perspective, identifying co-referent profiles must be derived from other meaningful sources that take into consideration the actual state and use of social networks, along with the content of interaction (whether textual or multimedia based). In Table 3.2, we summarize the previously proposed approaches for user profile matching and describe them with their features and limitations.

3.2.3 Discussion

While dealing with the task of profile matching seems to be easy in the presence of a global unique identifier, finding this global unique identifier remains challenging in the context of social networks. Until now, the previously outlined approaches didn't take into consideration the different types of data available on social networks. Entity resolution within the context of social networks requires, in one way or another, the use of different attributes extracted from users' profiles and data from users' interactions. Yet, multimedia data have not been exploited. However, the use of multimedia and its potential positive effect on entity resolution approaches is promising. It is therefore essential to harness the benefit of textual and multimedia data, i.e., from user profiles and from photos. This would be of great utility for finding and enriching links among same real-world persons.

Table 3.2: Summary of social network-oriented approaches for user profile matching

Approach	Features	Limitations
Ding et al. [138]	- Uses $foaf:mbox\_sha1sum$ and $foaf:homepage$ - Evaluates the importance of each FOAF attribute	- Matching profiles belonging to the same person fails when the IFP is different
Mika [139]	- Uses $foaf:mbox\_sha1sum$ and $foaf:name$ - Collects information from multiple sources and reasons based on name matching	
Golbeck et al. [140]	- Uses $foaf:mbox\_sha1sum$ and $foaf:knows$ - Implements an OWL reasoner to discover users having accounts on multiple social networks	
Bortoli et al. [141]	- Adds global unique identifiers into FOAF profile - Proposes a central repository to provide global unique identifiers for users	- Depends mostly on the user and the manual friend identification manipulation
Shi et al. [142]	- Compares entity labels and $foaf:mbox\_sha1sum$ - Describes two approaches to merge virtual identities: the first based on IFP, and the second based on heuristics	- No guarantee that two resources are the same when perfect label equality is used
Hogan et al. [143]	- Uses their own defined IFP - Proposes to partially identify equivalent entities from different sources in order to merge them	- Resources can exist without having a defined value for the IFP
Rowe et al. [144]	- Disambiguates persons by using social circles derived from social data - Makes use of different types of data (such as blogs, images, conversations, etc.) related to network users	- Selecting the identity features is done manually by the user
Zhou et al. [145]	- Applies logic and numeric methods - Combines sequentially two existing methods: an ontology-based social network integration approach and numeric reference reconciliation approach using semantic similarity	- Is not convenient when semantic structures become complex
Sleeman et al. [80]	- Describes a classification task using Support Vector Machine - Builds a classifier to identify co-referent entities using a supervised machine learning that uses a set of FOAF attributes	- Difficulties to build the classifier, learning data is limited and not extensible

As for enriching links with the type of social interaction, we present in the next section our review of different approaches used to identify link types.

3.3 Link Type Prediction

While most of the work in the area of social interaction has focused on predicting the existence of links [152] [85] [153] and measuring the strength of links between connected persons [154] [139] [155] [156], only few studies have focused on identifying the type of links connecting entities [157]. Link type prediction approaches, also called relationship discovery, can be categorized into three groups: 1) web-based, 2) domain-specific, and 3) photo-based. In the following, we detail these three groups of approaches.

3.3.1 Web-based Approaches

Social networks can be obtained from various sources, such as bibliographic databases, e-mail communications, and information extracted from the web. Most of web-based approaches [158] [159] [160] make use of the web to construct the social network. Then, these approaches proceed to identify the relationship types among persons.

In [160], a social network mining system called Polyphonet is proposed. It uses search engines to extract social networks and to infer the strength and the type of links among entities. Polyphonet can mainly identify four types of relations which are: *co-author*, *co-lab*, *co-project*, and *co-conference*. This is done by computing the number of co-occurrences between two persons' names on the web. To compute the strength between the two names, the number of returned hits/pages is measured. To infer the relationships' type, classification rules are obtained by manually assigning the correct labels to pages. Then, the system extracts the top ranked search results between two names and applies text categorization on those pages. At this phase, pages can be classified over classes of relations. Several indices, as stated in [160], are used to measure this such as matching coefficient, mutual information, Dice coefficient, Jaccard coefficient, overlap coefficient, and cosine.

In [158], the authors use search engines and mining techniques to extract social networks. Two social networks are used: artists of contemporary art, and famous firms in Japan. The relationship identification is based on the co-occurrence of names on the web. However, the authors argue that using only name co-occurrence becomes ineffective mainly in two cases: 1) when two entities co-occur universally on numerous web pages, and 2) when applied to

communities in heterogenous networks. To avoid this problem and to efficiently identify the relation between two firms for example, the authors propose to add a relation keyword to the search query to emphasize on a specific relationship. In the case of communities in heterogenous networks, such that of artists, a search query is made, and then an adaptive tuning of threshold is applied. This method takes into consideration the fact that, even though the relation is observed as weak, it might be important for the particular actors. Therefore, two criteria are implemented: 1) links are created when the link score is higher than a defined threshold, and 2) links are created based on subjective importance even if their score is less than the defined threshold. Hence, this satisfies the condition of having a number of links greater than a minimum number (isolated nodes) and lower than a maximum number (nodes with a big number of links).

The authors in [159] propose to detect the relationship type between named entities in an open domain where FOAF information and interpersonal relations are not available. The relation labeling starts by submitting two entities to a search engine and then by extracting cue patterns from the returned results. Cue patterns, also called local contexts, include person names, organization names, location names, and some other noun phrases. Then, each cue pattern is submitted to the Open Directory Project (ODP), a human edited directory, which will return the top N taxonomy paths. A directed graph is built using the returned taxonomy paths. After that, critical nodes or potential categories are extracted using a web page ranking algorithm such as PageRank, HITS, or Markov chain random process. At the end, most representative snippets are selected and used to extract potential relationships according to the defined critical nodes.

Most of the presented web-based approaches [158] [159] [160] proposed for relationship type discovery are really restrictive since they deal, in a way or another, with communities having information already published on the web (as most of the information publicly available on the web about persons is published for professional reasons). Hence, they can only be applied on communities in specific domains (e.g., researchers, famous firms, artists, etc.).

3.3.2 Domain-specific Approaches

In contrast with the web-based approaches which use the web to create the social network, the approaches presented in this category are applied on available social networks (e.g., researcher and political networks, email and mobile datasets, etc.).

In the work presented in [161], the authors propose a method to automatically extract the labels that describe relations among entities in social networks. The approach is applied on

a researcher and political social networks. To discover the type of relationship between two entities, the system extracts from web pages the surrounding keywords in which those entities co-occur. Common keywords are collected and ordered according to TF-IDF "Term Frequency-Inverse Document Frequency" [162] based scoring. Then, entities are clustered according to the similarity of their context. The representative terms in a cluster are extracted as labels to describe the relations of each entity pair in the cluster.

In the work of [163], a time-constrained probabilistic factor graph model (TPFG) is proposed to model the advisor-advisee relationship as a mining problem using a jointly likelihood objective function. The input of the graph model is the DBLP [148] bibliography database. First, a homogenous graph is constructed where an edge is created between every author's node and the publication node. Some assumptions based on common sense are used to remove unlikely relations of advisor-advisee. Two measures are used to reflect the correlation of the two authors and the imbalance of the occurrences between the advisor and the advisee. Within those two measures, the time factor is incorporated. Second, a dedicated learning algorithm is developed to infer the TPFG graph model. This graph is constructed to integrate local features like the relationship type and the start/end time of this relationship. By maximizing the joint probability of the factor graph, the relationship is inferred and the ranking score for each relation edge on the candidate graph is computed.

The authors of [164] address the problem of inferring social relationship in large networks. In order to learn how to infer social relationships ties, they propose a Partially-labeled Pairwise Factor Graph Model (PLP-FGM). The input of this graph model is partially labeled where some of the relationships are already known. In their work, they only focus on mining relationship semantics and do not consider content information. For instance, they apply their approach on three domains and consider the corresponding attributes: 1) a publication data network: *paper count*, *paper ratio*, *coauthor ratio*, *conference coverage*, *first-paper-year-diff*, 2) an email data set: traffic-based attributes, and 3) a mobile data set: number of *voice calls*, *messages*, *night-call ratio*, *call duration*, *proximity* and *in-role proximity ratio*. In order to infer relationship semantics, they consider three basic intuitions: user-specific information, link-specific information, and global constraints. In reality, these intuitions are a small number of rules that are manually devised by the authors for each data set domain. Their model learning is based on estimating a joint probability while maximizing the log-likelihood of observed information (labeled relationships).

To conclude, the approaches described in this section are applied on particular types of social networks. Although the extracted information from the web is domain depended, web

documents are highly heterogeneous, often unstructured, and may contain uncontrollable information that can make the mining process completely inaccurate (advertisements, popups, etc.) [161]. Furthermore, corresponding algorithms and rules, based on heuristics, are only applicable within a particular type of community [163] [164]. Manually devising rules is not an easy task mainly for two reasons: 1) generated rules don't consider the context of each user (published data, contacts with whom the user most interact, etc.), and 2) it is difficult to identify the extent to which rules are enough, complete, and correct for a given network domain (email networks, social networks, bibliographic networks, etc.).

3.3.3 Photo-based Approaches

Currently, social networks sites are great repositories of multimedia information, containing mainly photos and videos published by the users. For instance, more than 250 million photos[4], on average, are uploaded every day on Facebook. Approaches that propose to predict the type of relationship from users-generated data, especially from the profile information and the published photos, are still rare. The intuition behind this category of approaches is that when two persons are depicted together in a photo, this implies that they share some social connections.

The work presented in [165] is a preliminary investigation in measuring social networks through the natural social activity of being depicted in photos. A weighted graph is formed based on persons co-appearance information. The edge weight, between two persons, is computed in function of two parameters: 1) the number of photos in which the two persons appear (this directly affects the strength of the link), and 2) the number of persons depicted in a photo (this inversely affects the relationship strength).

In [166], the authors propose to detect social clusters from consumers' photos. A scheme is proposed to construct a weighted undirected graph by examining the co-occurrence of persons in photos. A closeness measure is defined here. It takes into consideration different elements in the photo: the distance between two faces, the number of faces, and the number of co-appearances in other photos. To detect the embedded social clusters, a graph clustering algorithm that maximizes the modularity of the graph partition is applied on the constructed graph.

The approach described in [167] proposes to discover the relationships (relatives, friends, child friends, and acquaintances) between the owner of a set of photos and the persons appearing within these photos. Two sets of rules are adopted here: 1) the first set of rules is qualified to be hard rules since related rules are considered to be always true, and 2) the second set of rules

[4]http://www.facebook.com/press/info.php?statistics - October 2011

is qualified as soft (since related rules are not always true) and generated from a training set (a photo collection where the relationship type is known). The inputs of this model are assumed to be provided by the owner of the photo collection. They are textual attributes and include the names of the persons, their ages, and genders. Learning on soft rules and inferring relationship types are based on a statistical relational model called Markov Logic.

The proposed approach by the authors of [92] estimates the relationship type among persons appearing in a collection of photos. From a family photo collection, a relation tree is constructed, and relationship types such as family, extended family, and friend are derived. To build the tree, the approach is mainly based on a face recognition algorithm, a clustering algorithm, and a face similarity algorithm. In this work, a gender classifier is trained and an age classification algorithm is developed to estimate the gender and the age of persons appearing in photos. In detail, the approach includes the use of age and gender, co-occurrences and relative position of persons, and photo timestamps. The relationship estimation starts by applying the face analysis technique to build clusters and to gather information from photos. Then, observations and expert knowledge obtained from experiments are applied to these photos in order to identify the relationships.

The described approaches [165] [166] [167] in the photo-based category are interesting but suffer of several drawbacks. First, they only consider one subset of attributes available in photos. The input is mainly restricted to the following attributes: age, gender, and co-occurrences of persons. Considering only those attributes would obviously limit the capacity of the system to exploit other rich information available, to enhance, and to facilitate the discovery of all relationship types. Second, users' profile, or the profile of the owner of the photo album, is not taken into consideration. Third, they are not user-based. In [167], the authors define an interesting approach to discover relationships between persons within photo collections. Although the approach defines a set of hard and soft rules to discover the relationship types, it assigns all the soft rules with the same weight regardless of the users' context. Consequently, this approach is not able to adapt to users by considering their photo collections. Last but not least, the work described in [92] is only restricted to visual features.

3.3.4 Discussion

Social networks have become a large repository of information, connecting persons that share together different kinds of social relationships (colleagues, relatives, and friends). Through users' interactions on social networks, different explicit and implicit data emerge (e.g., messages, posts,

common friends, photos, comments, etc.). Previously, different textual attributes have been considered. However, multimedia data, such as photos, could be of great importance for the link type prediction task. In fact, photos capture natural and everyday human interactions, thus are of valuable consideration for link type prediction tasks. So far, this fact is ignored [158] [159] [160] [161] [163] [164]. Consequently, these approaches are missing crucial information for the task of link type prediction. Meanwhile, some other works tried to incorporate photos in their approaches [165] [166] [167]. Nonetheless, current photo-based approaches consider only a small subset of attributes to extract from photos. Although these approaches tried to analyze the semantic features from photos by issuing common sense based rules, their rules are manually predefined and not extendable. Building up new link type prediction approaches, regardless of the type of social network and relationships, is yet a challenging open issue. In Table 3.3, we summarize the photo-based relationship discovery approaches and compare their characteristics.

3.4 Conclusion

Social networks are considered as a useful tool for social matching where the provided personal information by the users can be exploited to obtain better profile matches. As for social relationship discovery, the interaction on social networks between users is of great importance as a potential way for building interpersonal relationships and group ties. Matching co-referent users and discovering the type of these interpersonal relationships are the two parts of the issues that we address in this work. In this chapter, we presented the major works related to these two topics. The findings of our review reveal that:

- **User profiles' attributes:** Current entity resolution and link type prediction approaches are not well adapted to the actual social networks sites. Exploiting in depth all the information available in attributes of user profiles seems inevitable.
- **Photos:** Photos are increasingly gaining importance due to their popularity. Setting the focus on photos is of considerable importance for extracting essential evidences of users' interactions.
- **Framework:** Methods to enrich the links' characteristics are diverse and disparate. Currently, their is no unique framework able to deal with both entity resolution and link type discovery tasks.
- **Methodology:** Methodologies describing how to learn rules in order to identify co-referent

Table 3.3: Summary of photo-based approaches for link type prediction

Approach	**Used attributes**	**Visual features**	**Relationship**		**Based on**		**Extended rules**	**Users' contexts**	
			Type	Closeness	Rules	Graphs		Mined rules	Rule's weight
Golder et al. [165]	frequency of co-occurrence	no	no	yes	no	yes	-	-	-
Wu et al. [166]	co-appearance, face distance, number of person in a photo	yes (face detection and clustering)	no	yes	no	yes	-	-	-
Singla et al. [167]	age, gender, co-appearance	no	only one	no	yes	no	yes	no	no
Zhang et al [92]	age, gender, co-appearance, time	yes (face analysis)	only one	no	yes	no	no	no	no

users and infer relationships types are missing. Nonetheless, we believe such novel methodologies to enrich social networks' links are of considerable interest.

- **Context:** Data describing a user profile differs among users and across social network sites. It is obvious that the design of reliable methodologies must take into consideration data related to the context of each user/site.

It is therefore crucial to propose approaches that provide new solutions for entity resolution and link type discovery. We believe that investigating these two topics together will serve as a base for future applications that require semantically enriched social networks links. In this study, we address these challenges and present our approach for discovering relationships' types among users within a single social network and to find co-referent users across different social networks.

Chapter 4

Framework

ABSTRACT

In this chapter, we present our framework for semantically enriching links among social network users. We describe all the components that are used by this framework. We also propose a data model that gives a coherent view of different concepts and various types of data available on social networks. Concepts, represented using appropriate standards and ontologies, are interrelated together as a way to model users' interactions over social networks. This data model is used to extract, analyze, and infer new knowledge from implicit and explicit data using a provided set of functions.

4.1 Introduction

The volume of information available on social networks is exponentially growing at an incredible rate, going beyond social network users' ability to easily interact and manage it. Social network users encounter data in different situations - whether while checking users' profiles or browsing online photo albums, reading comments or replying to messages. Sometimes, this data is explicit (e.g., information from users' profiles); other times, it is implicit. Users' activities and interactions over time are among the most important sources of implicit information. These online activities and interactions come in many guises, i.e. messages, photos, comments, etc. For instance, some real applications build up on implicit information discovery, and use it in various ways to: construct the graph of message communications between a group of users, extract date and time information from photos to build a timeline photo gallery, or retrieve GPS location from photos in order to indicate on a map all the cities that a social network user has visited, etc.

Explicitly available data is only the tip of the online social networks iceberg. With the current rise of social networks, explicit information can be obtained from users themselves (information available on their profiles), whereas much more efforts must be invested to gather implicit information. While explicitly available information is a valuable resource, implicit information can have important implications towards enhancing the understanding of social interactions. Dealing with explicit as well as implicit data is part of this work.

Our ultimate goal is to semantically enrich links among users within a single social network and across several ones. In terms of understanding social interactions, much more efforts have been invested in identifying relationships between persons primarily through the use of explicit information (e.g., information from search engines or from available data sources) and, to a lesser extent, through the use of implicit information extracted from various sources (particularly from multimedia data). In this chapter, we focus on the use of explicit and implicit data to give a richer understanding of situations involving persons' interactions.

Actually, gathering information on social networks involves identifying two main points: 1) the implicit information to be inferred, and 2) the resource or the activity from which this information can be extracted. Thus, the initial phase of our work is to gather these information. To this end, as illustrated in Figure 4.1 we consider three main sources of data:

- **Service data:** all information provided by users to the social network site.

  Service data exists within profiles that contain information explicitly describing users. Different technologies provide users with an extensive list of attributes to describe their pro-

files: Resource Description Framework - in - attributes (RDFa)[1], Microformats[2], XHTML Friends Network (XFN)[3], and Friend Of A Friend (FOAF)[4]. The choice of the attributes that describe users' profiles relies primarily on the social network site. Users profiles' attributes usually contain information related to the demographics of users (e.g., age and gender), users' personal and professional addresses, the interests of users, users' preferences, etc. In the context of social networks, users can create several profiles, each on a different social network site. In such case, each profile may have different representations on disparate social network sources. It is therefore necessary to have a common profile representation that would allow both of the identification of relationship type between users, and the identification of the multiple profile occurrences.

- **Disclosed and Incidental data:** While disclosed data results from users' online activity, the incidental data results from the activity of contacts on users' profiles.

  With the increasing popularity of photo sharing sites, significantly more information can be extracted from photos shared by users. It is therefore interesting to extract disclosed and incidental data from the set of photos shared by users. Photos capture natural human interactions and contain either explicit or implicit contextual information. In fact, context in which photos are captured is of considerable importance since social relationships between users can be implied from photos.

  The term *context* has many definitions in the literature and across different areas in computer science. One commonly used definition is given by the authors of [168] who defined the context as: "Any information that can be used to characterize an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves". They also stressed on the importance of the following four aspects of context: location, time, activity, and identity.

  Explicit information from photos can include the captured date and time of a photo, the location where the photo was captured, the name of the persons depicted in a photo as well as their estimated age and gender (if needed). As for implicit information that can be inferred from photos, it is possible for instance to infer information illustrating, to main users, persons with whom they usually co-appear, the age and the gender categories of these persons, and how they are related to other contacts, etc.

[1]Resource Description Framework-in-attributes, http://www.w3.org/MarkUp/2009/rdfa-for-html-authors/
[2]Microformats, http://microformats.org/wiki/Main_Page/
[3]XHTML Friends Network **(XFN)**, http://gmpg.org/xfn/
[4]Friend Of A Friend **(FOAF)**, http://xmlns.com/foaf/spec/

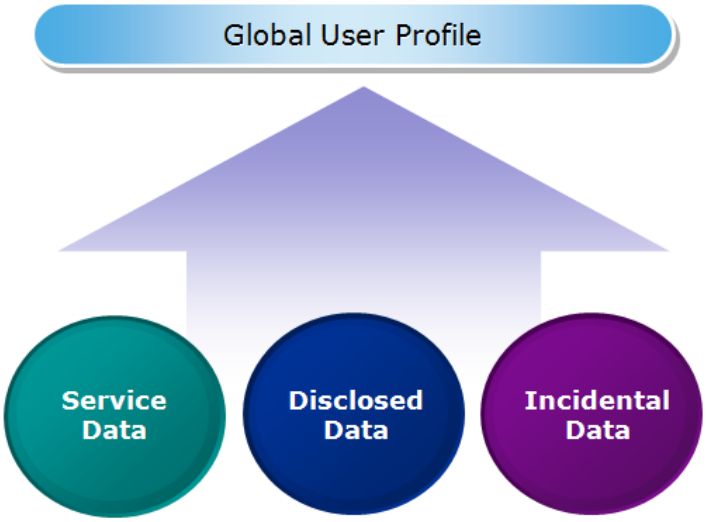


Figure 4.1: Different types of data used to form a global user profile.

We believe that enriching links among social network users could greatly benefit from the analysis of both explicit and implicit data gathered from users' profiles and extracted from social network activities. We, therefore, propose a framework and provide a data model appropriate for: 1) discovering the type of social relationships that main users share with their contacts within one social network, and 2) matching profiles related to a main user and that refer to the same real-world contacts across different social network sites.

In our approach, we combine the profile information extracted from users' profiles (personal and professional information) with the contextual information extracted from available photos (e.g., the date, the time, and the location of the captured photos, etc.). Since it is important to accommodate our approach to different social network users having different behaviors, we designed a context-aware framework able to put users at its heart. With the use of a set of generated rules, as presented in the next chapters, we propose to discover the type of social relationships and to match co-referent users across social networks. We categorize the rules as: *intra-social* rules, when these rules target the relationship discovery task within a single social network site; and *inter-social* rules, when they target the task of identifying co-referent users across different social network sites.

Before describing our approach in the next chapters, we dedicate this chapter to present our framework and adopted data model.

4.2 Framework

The input of our approach is composed of the set of main user's contacts with their profiles and shared photos retrieved from a single or many social networks. The proposed framework contains six main modules, as illustrated in Figure 4.2:

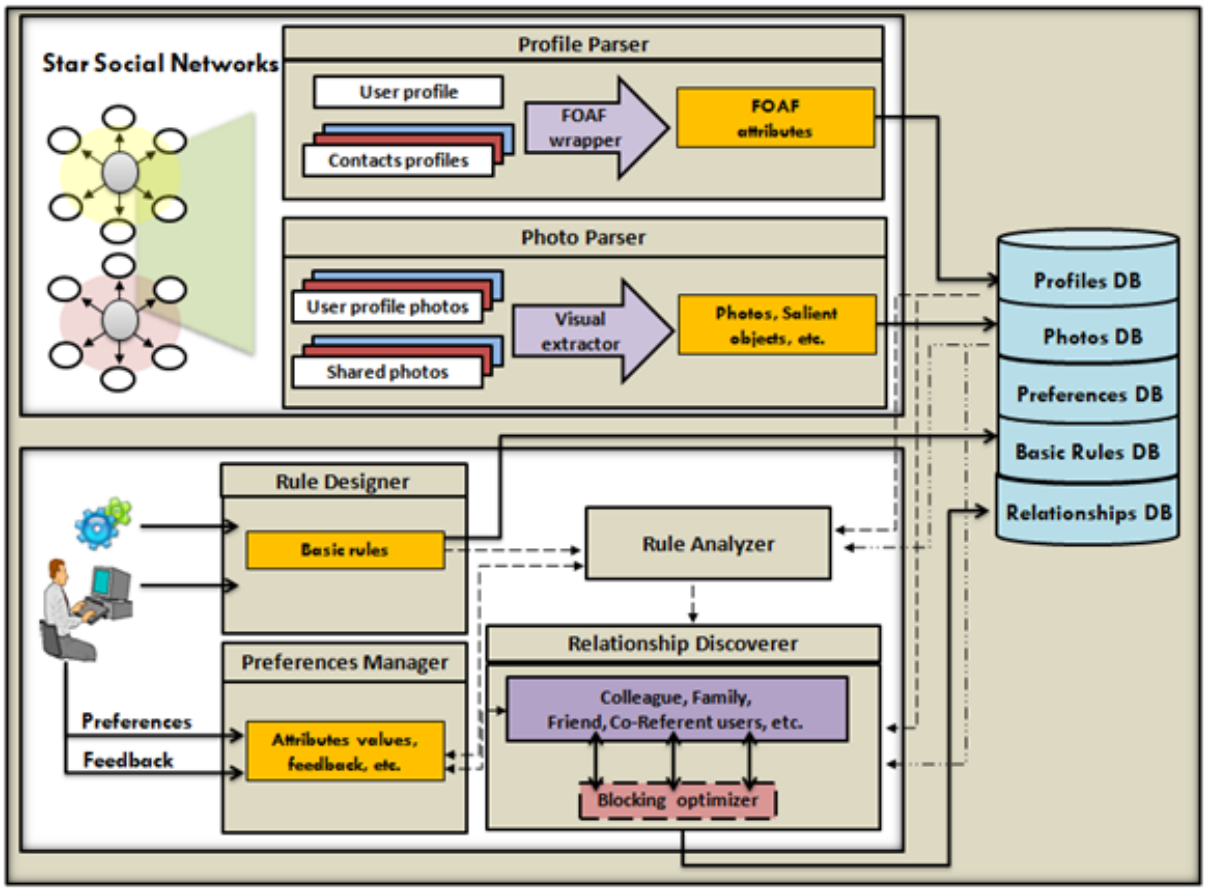


Figure 4.2: Architecture of our framework to semantically enrich links.

1. **Profile Parser:** extracts the profile of the main user as well as the profiles of all available contacts (e.g., age, gender, working place name, school name, etc.). The values of all the selected attributes are stored in the profile database (Profiles DB) component.

2. **Photo Parser:** retrieves, analyzes, and stores the photos of the user's profile as well as all the photos shared by the user's contacts in order to extract available metadata, captions, descriptions, tags, and other information defined in our data model (c.f. Section 4.3).

3. **Preferences Manager:** provides the main user with the possibility to personalize some

preferences attributes. User's preferences can then be accommodated to the set of relationships used to enrich links, namely social relationship types and co-referent links. Since applying rules may need to consider user's feedback, the preference manager module enables the user to refine the used rules. As detailed later on in this chapter, preferences are divided into three groups:

(a) General preferences: common preferences related to the user's basic settings such as relationships to discover, age category settings, and the main user's age category.

(b) Relationship-based preferences: each social relationship can have its own preferences such as day, time, location, and predefined keywords.

(c) System preferences: matching co-referent users can be personalized using the weights of profile's attributes, classical unique identifiers, similarity metrics, and aggregation functions.

4. **Rule Designer:** contains the set of basic rules used to enrich the links either with social relationship types or with co-referent links. These rules use attributes related to users' profiles and to photos. We present them in the next chapter.

5. **Rule Analyzer:** applies the set of basic rules on the whole set of photos and profiles. Since a number of rules may be missing in the set of basic rules, we use mining techniques [12] to extend this set. It is worth noting that different users may get different extended rules since the mining takes into consideration available information extracted from the profile of each user.

6. **Relationship Discoverer:** is responsible of assigning the corresponding relationship(s) to users. A social relationship (such as colleagues, relatives, friends, etc.) is assigned to a main user and her contacts within a single social network. A co-referent relationship is used to identify co-referent contacts connected to a main user across different social networks. This module communicates with the "Blocking optimizer" which is used to enhance the performance of the "Relationship Discoverer". The "Blocking optimizer" module aims to reduce the number of profile comparisons for the task of co-referent users identification.

In the next section we present our data model for representing social network users and their data. These represented concepts are used by different modules of our framework which requires the extraction of explicit and implicit data in order to semantically enrich links between social network users.

4.3 Data Model and Definitions

In order to gather and exploit data from social networks, we describe our data model used to represent profiles and other concepts part of our approach.

4.3.1 User Profile

User Profile allows to describe the characteristics of a user in the social network. It is defined as follows:

User Profile: (profileID, FOAF_attribute)*

where:

- *profileID* is the identifier of the owner of the profile.
- *FOAF_attribute* is the set of attributes used to describe the profile of a user. These attributes are part of the FOAF vocabulary.

4.3.1.1 Why FOAF?

Currently, social network sites do not all adopt the same user profile attributes' representation. As mentioned previously, there are different ways to represent the attributes of profiles using different vocabularies, including Resource Description Framework - in - attributes (RDFa), Microformats, XHTML Friends Network (XFN), and Friend Of A Friend (FOAF).

- **Resource Description Framework - in - attributes (RDFa):** is a W3C recommendation used to embed semantic into XHTML. RDFa is a thin layer of markup that can be added to web pages and make them more understandable for machines as well as for persons. RDFa provides a consistent syntax and big expressivity by proposing an integration of the RDF triple concept (subject, predicate, attribute) with the flexible XHTML language, which is used by web browsers.
- **Microformats:** are little pieces of structured information embedded into XHTML documents. They transform documents to machine-readable semantic data such as contact details, social relationships, event information, etc. Currently, different microformats exist for different needs such as hCard used to describe persons, companies, and organizations

with a limited set of elements representing business cards, calendars for events (e.g., hCalendar), decentralized tagging (e.g., rel-tag), etc.

- **XHTML Friends Network (XFN):** represents 18 human relationships with a set of values and gives the possibility to authors, for example, to indicate which of the weblogs they read belong to friends they have met.
- **Friend Of A Friend (FOAF):** is a machine-readable semantic vocabulary describing persons, their relationships, and activities. It defines a set of attributes, grouped into categories as shown in Figure 4.3. FOAF documents are written in XML syntax and adopt the conventions of the Resource Description Framework[5] (RDF).

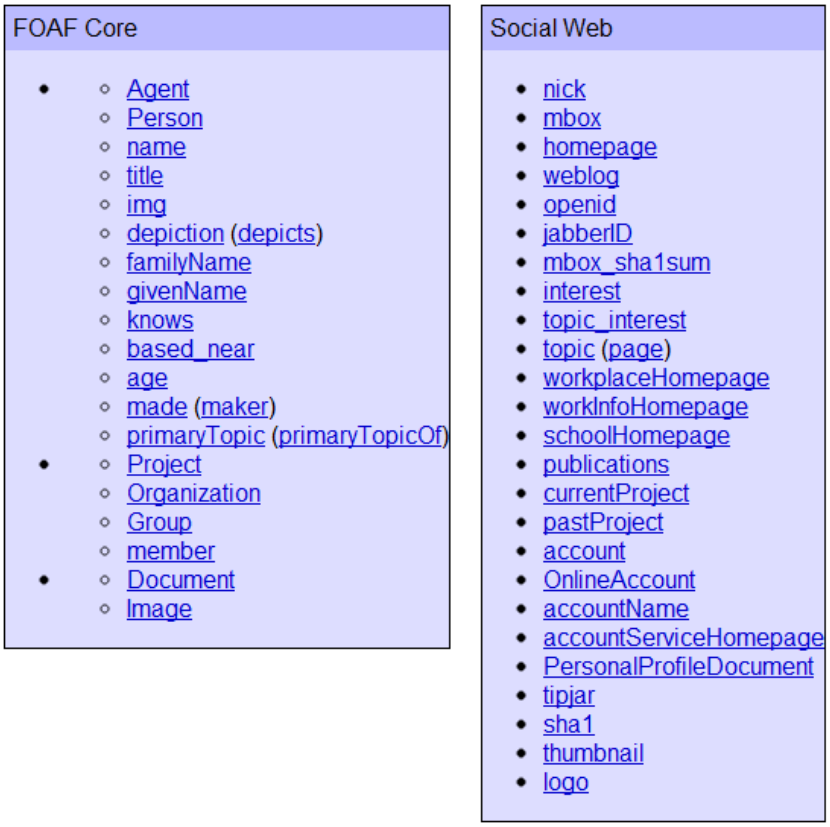


Figure 4.3: FOAF attributes.

In this work, we adopt FOAF, among many other vocabularies (e.g., Resource Description Framework - in - attributes (RDFa), Microformats, XHTML Friends Network (XFN), etc.), to represent a social network profile. Different attributes can be used to describe profiles such as

[5]Resource Description Framework **(RDF)**, http://www.w3.org/RDF/

```
<foaf:Person>

<foaf:name>Alexandre William</foaf:name>
<foaf:firstname>Alexandre</foaf:firstname>
<foaf:family_name>William</foaf:family_name>
<foaf:mbox rdf:resource=aw@somesite.com/>
<foaf:homepage rdf:resource=http://personalsite.com/aw/>
<foaf:workplaceHomepage rdf:resource=http://workaddress.com/>
<foaf:img>www.xyz.com/alex/photos/alex.jpg</foaf:img>
<foaf:interest>Paris, Software, Internet</foaf:interest>

<foaf:knows><foaf:Person>
<foaf:mbox rdf:resource=contact1@somesite.com />
<rdfs:seeAlso rdf:resource=http://contact1.net/foaf.rdf./>
</foaf:Person></foaf:knows>
<foaf:knows><foaf:Person>
<foaf:mbox rdf:resource=contact2@somesite.com />
</foaf:Person></foaf:knows>


</foaf:Person>
```

Figure 4.4: A simple profile represented using FOAF vocabulary.

name, *image*, *projects*, *work*, etc. In reality, FOAF is considered as the richest vocabulary to use in terms of describing users' profiles and has currently become a widely accepted standard since many large social networking websites propose a FOAF profile for their users [140]. Nowadays, FOAF is admitted to be one of the real success story of the semantic web [169]. A FOAF example is provided in Figure 4.4.

Although FOAF is a rich vocabulary for describing users, it is not suitable for the description of other data types (photo albums, photos, and salient objects). Therefore, these data types and how we represent them are detailed later on.

4.3.2 Star Social Network

Star Social Network is a star graph[6] with a group of users connected to a given user u_0 using a set of relationships. In our study, we define the star social network as follows:

$$G: (U, Profile, R, \mu, \alpha)$$

[6]It is a directed graph when its links connect a given user u_0 to her contacts and it is an undirected graph when its links connect contacts of u_0 across different social networks

where:

- U is a set of users.
- *Profile* represents the profile(s) of each user. A user can have more than one profile on a single social network or across different ones.
- R contains the set of relationship types adopted in the social network. These relationships can be categorized into two groups: 1) **social relationships** (such as *friends*, *colleagues*, *relatives*, etc.) and 2) **co-referent relationship** (namely *SameAs*) to refer to profiles that represent same real-world persons.
- μ is a function that assigns each user u_i to her corresponding profile(s) with a single or across different social networks.
- α is a function that assigns the relationship R_i between:
  - u_0 and u_i: α assigns a social relationship type r between the main user u_0 and other users within a single social network. For instance, friends(*Alice*, *Dupont*) means that the given user *Alice* and *Dupont* are friends. Here, social relationships are dependent to the given user and thus directed accordingly (as illustrated in Figure 4.5).
  - u_i and u_j: α identifies co-referent users and assigns a co-referent relationship type between two users u_i and u_j across different social networks. For instance, SameAs(*Dupont*,*Dupond*) means that two users *Dupont* and *Dupond* refer to the same real-world person. Here, co-referent relationships are between the contacts of a main user (as illustrated in Figure 4.6).

In this work, we assume that a user can have only one profile within a single social network. We note by G_r a sub-graph of G where only the relationship type r is used between u_0 and her contacts:

$$G_r\text{: (U, Profile, \{r\}, } \mu\text{, } \alpha\text{)}$$

4.3.3 Photo

Photo contains embedded attributes with descriptions that illustrate and give important information about the captured scene. In this study, we only focus on photos that depict at least one person within the captured scene(s). More formally, we represent a photo in the following way:

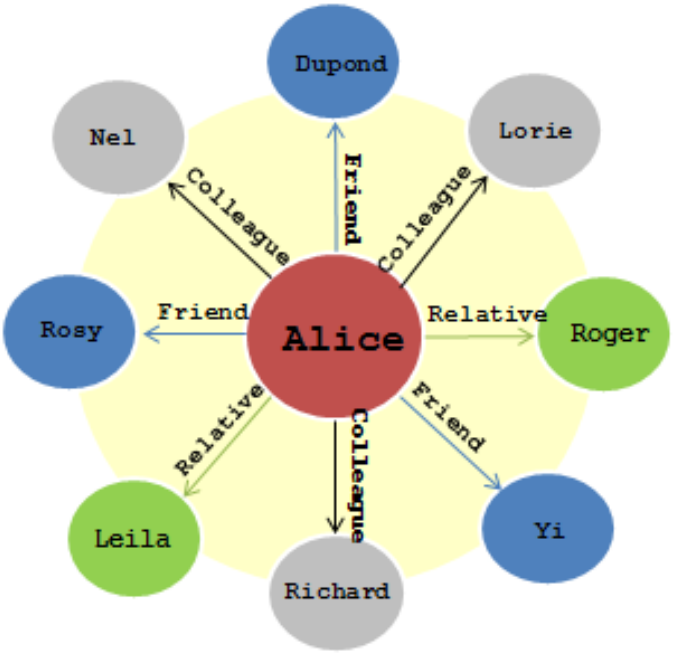


Figure 4.5: A sample star that represents the personal social network (PSN) of Alice with three relationship types.

Photo: (pID, publisher, meta, caption, comment, so*)*

where:

- *pID* represents the photo Universal Resource Identifier (URI).
- *publisher* is the person who published the photo. This person can be the main user.
- *meta* or *metadata* is the set of technical descriptions embedded in the photo that takes the form of data about data. Here, metadata can be defined as information not related to the semantic content of the photo, i.e., author, location, date, etc. In fact, adding metadata at the photo creation time requires no effort. Actually, recent digital devices provide such functionality. This is considered as a best practice if all the information related to the date, the place, and other technical characteristics are available when capturing the photo. Here, we assume that photos use the Exif[7] standard to extract the following metadata in a photo: Date, Time, and Location.
- *caption* is the textual description provided by the publisher of the photo to describe the semantic content of a photo. Usually, a caption gives some relevant information about the

[7]The Exchangeable Image File Format **(EXIF)**, http://www.exif.org

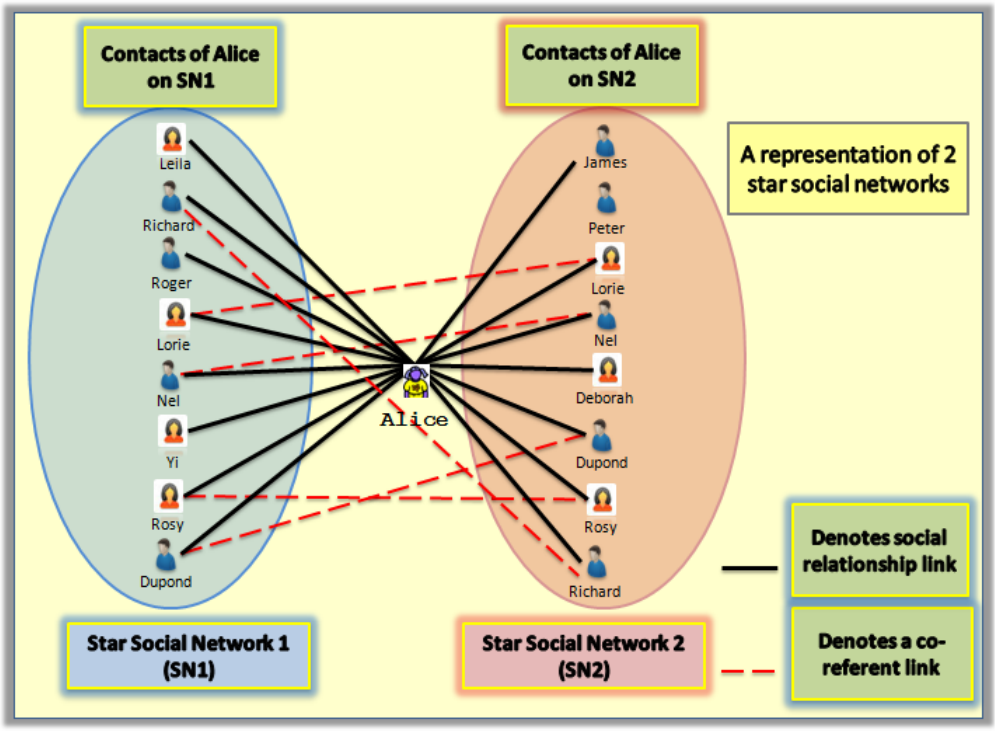


Figure 4.6: A sample star social network that shows the contacts of Alice on two star social networks.

photo in terms of places, objects, persons, etc. This could be useful and helpful in the process of discovering the type of relationship between users. We adopt the Dublin Core[8] (DC) metadata standard to represent the caption related to a photo.

- *comment** is the set of comments that can be added to a photo either by the owner of the photo or by her contacts. We adopt the DC metadata standard to represent the comments related to a photo. A comment is defined as follows:

  Comment: (cID, text, profileID, date/time)

  where:

  - *cID* represents the comment identifier.
  - *text* is the textual description of the comment added by either by the owner of the photo or by her contacts.
  - *profileID* is the profile identifier of the person who adds the comment.

[8]The Dublin Core Metadata Initiative **(DCMI)**, http://dublincore.org

- *date/time* is the creation date and time of the comment.

- *so** is the set of salient objects depicted in the photo.

4.3.3.1 Why Exif?

Exif is well-suited for the representation of digital images where the focus is on the technical aspects of the metadata. We choose Exif for its simplicity and since most of digital cameras store automatically such information in the header of the images using this standard. In Exif, captured images are described using a predefined set of attributes in terms of the context, the environment, and the technical characteristics (exposure time, resolution, focal length, date/time, GPS location, etc.). In this work, we are interested in metadata related to the location, date, and time of the photo capture. The value of the *Location* metadata is extracted from the following attributes: *exif:gpsLatitude*, and *exif:gpsLongitude*. The values of the *Date*, and the *Time* metadata are extracted from the following attribute: *exif:dateTime*.

4.3.3.2 Why DC?

We adopt the DC metadata standard to represent the caption and comments related to a photo. DC is appropriate to represent the different kinds of textual descrptions. It is a compact metadata standard used for cross-domain information resource description. It is an XML/RDF based syntax, hence it is useful when disparate metadata formats are used together. DC is simple and easy to implement thus allowing non-specialists to use it (e.g., librarians, researchers, museum curators, music collectors, etc.)

4.3.4 Salient Object

Salient Object represents the face of a person in a photo. In our study, a photo can contain at least one face, or salient object (*so*), that must have a match with one of the profiles of the social network in order to be identified. Faces are particularly important in our model not only to identify the related persons (when applying face detection and recognition algorithms) but also to extract other interesting information related to the age and gender since several visual (age and gender) estimators can be applied [170] [171].

Two main methods are possible to get *so**: 1) Manual: some social networks (such as Facebook, MySpace, etc.) provide users with the possibility to tag salient objects in photos (by draw-

ing rectangles on the related regions and filling the names of corresponding contacts/persons), 2) Automatic: using face detection and face recognition algorithms to detect and identify persons in photos. In this work, we assume that *so** are already identified and tagged by users (method 1).

Formally, a salient object, illustrated in Figure 4.7, is defined as:

so: (soID, photo, soPublisher, taggedPerson, comment, date/time, location)

where:

- *soID* is the identifier of the given *so*. A *so* can represent the owner of the photo or one of her contacts.
- *photo* is the photo to which the *so* belongs.
- *soPublisher* is the person who posted the *so*.
- *taggedPerson* is the person identified in the *so*.
- *comment* is an annotation text, also called tag, assigned to this salient object by the owner of the photo or one of her contacts. The *so*'s creator, date/time of creation, and the text of the tag are represented using the DC metadata standard.
- *date/time* is the creation date/time value of the *so*.
- *location* is the region containing the salient object in the photo. The salient object is not assigned a fixed width and height. Its coordinates determine the location and the size of the salient object region. We represent a *so* as a region in the photo using the MIRO[9] ontology.

4.3.4.1 Why MIRO?

MIRO describes what is depicted within various types of multimedia objects (including image and videos). Depicting a region within an image can be easily achieved using the MIRO ontology. It is a simple way to locate, within the image, a region delimited by its coordinates.

[9]Mindswap Image Region Ontology **(MIRO)**, http://www.mindswap.org/2005/owl/digital-media

Figure 4.7: Sample photo (with its salient objects) shared on the social network of Alice. The information displayed in each of the two boxes refers in the order to the identifier of the so, the identifier of the photo on which the so is published, the profile identifier of the person who published the so, the profile identifier of the person who is tagged, the text used to tag the person, the date/time of the tag publication, and the coordinates of the so region within the photo.

4.3.5 Photo Album

Photo Album is a set of photos grouped together. A wide practice consists of grouping photos of the same event in one photo album. In addition to the photo albums of a given user u_0, other users can also share their albums with u_0. Photo albums' have the values of their attributes (title, descriptions, and comments) common to all their photos. These values are given by the creator of the photo album.

An album is represented as follows:

PhotoAlbum: (aID, publisher, title, desc, comment, photo*)*

where:

- *aID* represents the photo album Universal Resource Identifier (URI).
- *publisher* is the creator of the album.

- *title* is the title of the album represented using DC.
- *desc* is the description assigned to the whole album by the creator of the album.
- *comment** is the set of comments that can be added to an album either by the creator of the album or by her contacts. For each comment, the identifier of its creator and the date of its creation are stored.
- *photo** is the set of related photos. Photos can be added by the creator of the album or by one of her contacts.

DC is used to represent the creator, the date/time of creation, and the text description of the *desc* and the *comment* of the photo album.

4.3.6 User Preferences

User Preferences enable the user to personalize the preferences to suit her choices. User preferences are stored in the database (Preferences DB component). We distinguish three groups of preferences:

1. **General Preferences:** these preferences are common preferences related to the user's basic settings.
   (a) *relationships*: the set of relationships to be discovered,
   (b) *age categories and their ranges*: persons with different ages are assigned to user-defined categories such as for the user-defined age categories. By default, we consider the following categories: child (age $\leq$ 13), teenager (14 $\leq$ age $\leq$ 18), adult (19 $\leq$ age $\leq$ 60), and senior (age $\geq$ 61),
   (c) *main user's age category*: the category to which the main user belongs.
2. **Relationship-based Preferences:** these preferences are defined for each relationship (colleagues, relatives, friends, etc.)
   (a) *date* and *time*: date and time related to each relationship type. For instance, a main user can define her working/not working hours and working/not working days.
   (b) *location*: GPS location related to each relationship type. It can refer to the GPS location of the user's house, or the user's work place.

(c) *predefined keywords:* the set of keywords that according to the user can be used to identify a relationship type.

3. **System Preferences:** these preferences refer to some advanced settings that the user can modify. We provide default values for all the system preferences, however, users can change the default settings. The following preferences are used to assist a main user in the task of co-referent identification:

   (a) *classical unique identifiers:* the default unique identifier that we use to refer to co-referent users is *foaf:mbox_sha1sum*.

   (b) *weights of profile's attributes:* for the task of profile matching, the weights of attributes can be assigned manually and/or automatically. As it will be detailed in Chapter 5, we propose to compute automatically these weights, however, it is possible to modify them manually.

   (c) *similarity metrics:* the list of similarity metrics we use for different types of attributes. These metrics are detailed later on in this chapter.

   (d) *aw:* the aggregation functions used for decision making. They are detailed later on in this chapter.

These preferences are defined as:

Pref: (prefID, type, attribute, value, relationship)

where:

- *prefID* is the identifier of the preference.
- *type* represents the type of the preference where *type* $\in$ *{General, Relationship, System}*.
- *attribute* is the attribute related to a given preference type as defined previously.
- *value* is the value that the user assigns to the attribute.
- *relationship* indicates if the preference is specific to a defined relationship type such as colleagues, relatives, and friends, etc. It is only required for relationship-based preferences.

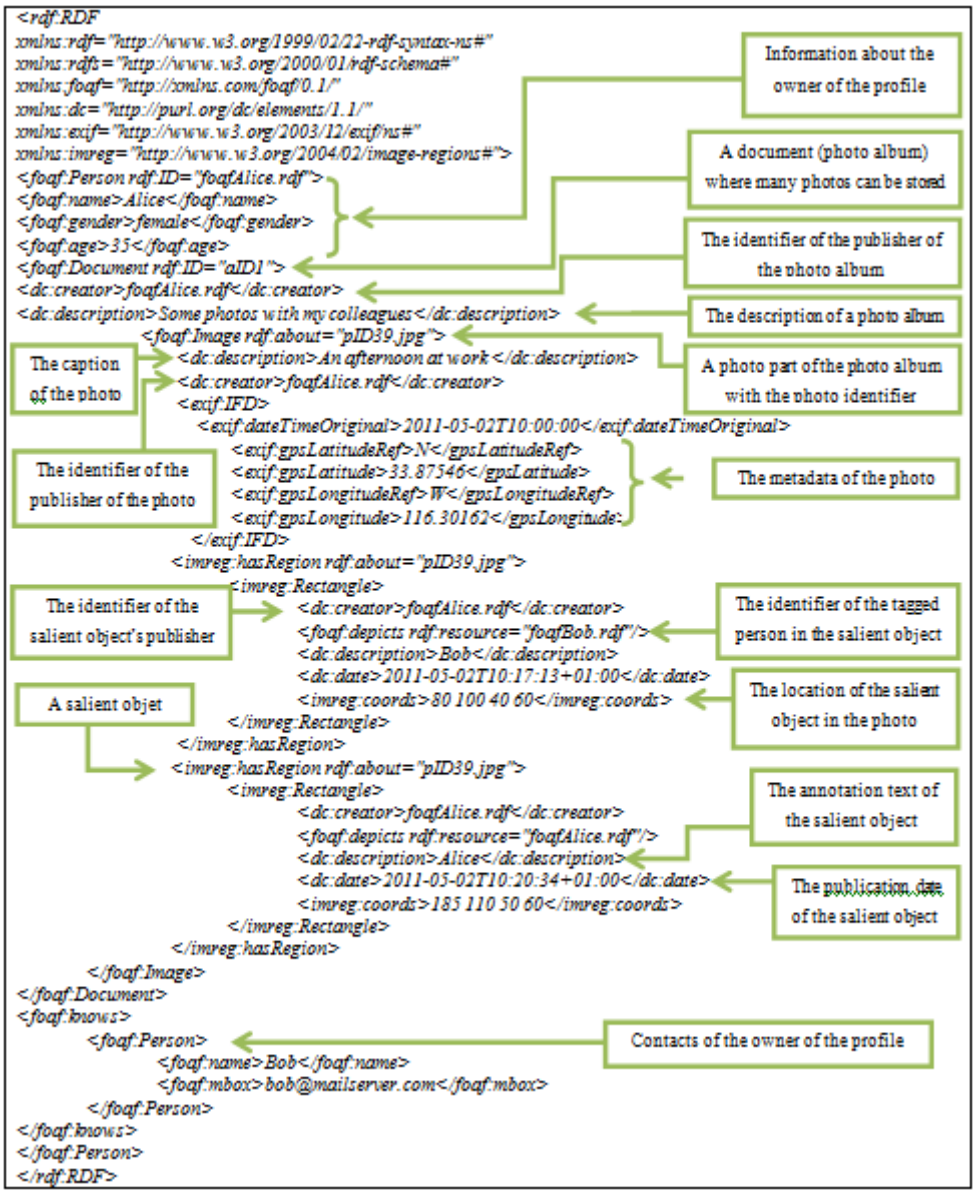


Figure 4.8: Extract of the global profile of Alice using our model.

Example:

To illustrate the proposed data model used in this study, let us consider the shared photo of user called Alice in Figure 4.7. The user profile of this user, represented in RDF syntax, is described in Figure 4.8. For the sake of space reduction, we only display a small set of information available in her profile, one photo album, and one of her contact list. In this example:

- **User Profile:** the user profile of Alice displays some personal information using FOAF

(*foaf:name*, *foaf:gender*, *foaf:age*). In this user profile, it is indicated that Alice knows another person (*foaf:knows*). The name of this person (*foaf:name*) is Bob and he is described with his email address (*foaf:mbox*).

- **Photo album:** In addition to these personal information, a photo album (*foaf:document*) is described in the profile. The photo album's publisher (*dc:creator*) and photo album's description (*dc:description*) are represented using DC in the user profile. This photo album, identified with an album identifier (*aID1*), can contain a set of photos.

- **Photo:** In our example, a single photo (*foaf:image*) is displayed and described with a text (*dc:description*). Its metadata related to the date, the time, and the GPS location of the photo capture are described using Exif (*exif:dateTimeOriginal*, *exif:gpsLatitudeRef*, *exif:gpsLatitude*, *exif:gpsLongitudeRef*, *exif:gpsLongitude*). The photo's publisher (*dc:creator*) and the photo's identifier (*pID39.jpg*) are indicated as well.

- **Salient objects:** The photo depicts two salient objects that represent two persons within the star social network of Alice. The location of each of the two salient objects is described with MIRO (*imreg:hasregion*). The salient object is described with the following attributes: the identifier of the person who tagged the photo (dc:creator), the identifier of the tagged person (*foaf:depicts*) in the photo, the textual description of the tag (*dc:description*), the date of creation (*dc:date*), and the tag region's coordinates (*imreg:coords*).

4.3.7 Semantic Rules

Applying rule-based reasoning can be used to discover relationships while having rules typically expressed as a set of conditions. **SemanticRule** enriches the links semantically within a star social network either with social relationships or co-referent relationships. A rule takes the form of:

$$SR:\ body \longrightarrow \langle head, score \rangle$$

where:

- SR is the name of the semantic rule.

- *body* is a boolean expression written as a conjunction of conditions. These conditions rely on a set of functions related to *user profiles*, *photo albums*, *photos*, and *user preferences,* as detailed in the next section.

Table 4.1: Example rules used in this work to discover social relationships and identify co-referent contacts. Each semantic rule has a name and assigns the relationship type of the rule's head when all the conditions of the rule's body are satisfied.

SR1:
pID.GetKeywords("description").TextSimilaritymetric(u0.Preference.GetRel- Keywords("friends")) >= 0.9
→ (**friends**, 1)
SR2:
(pID.GetPhotoCreationTime() == u0.Preference. GetWorkTime()) AND
(pID.GetPhotoCreationDate() == u0.Preference. GetWorkDays()) AND
(pID.GetPhotoLocationGPS() == u0.Preference.GetLocationGPS("colleagues"))
→ (**colleagues**, 1)
SR3:
(pID.GetNumberPersonsPhoto() == 2) AND
(profileID1.CoAppearWithPerson(profileID2) > 1) AND
(profileID1.GetAge() == u0.Preference.AgeRangeDist().child) AND
(profileID2.GetAge() == u0.Preference.AgeRangeDist().adult) AND
(pID.GetPhotoCreationTime() == u0.Preference.GetNotWorkTime()) AND
(pID.GetPhotoCreationDate() == u0.Preference.GetNonWorkDays())
→ (**relatives**, 1)
SR4:
(profileID1.name.TextSimilaritym(profileID2.name) >= 0.8) AND
(profileID1.lastname.TextSimilaritymetric(profileID2.lastname) >= 0.7) AND
(profileID1.nickname.TextSimilaritymetric(profileID2.nickname) >= 0.9)
→ (**co-referent**, 1)

- *head* holds the relationship (such as colleagues, relatives, friends, co-referent, etc.) between users if the body part is evaluated as true.

- *score* $\in [0, 1]$ is the weight assigned to a rule. It therefore allows to prioritize rules' importance or filter rules accordingly.

For instance, let us take the semantic rules listed in Table 4.1. These semantic rules are assigned a name, a body, a head, and a score. In these examples, the score is equal to 1 which indicates that these rules have the highest importance compared to other rules. In the following we explain the objective of each semantic rule:

- SR_1 : The relationship type $\langle Friends \rangle$ is assigned to the persons depicted in the photos when at least one of the photo's description keywords and the friends' keywords indicated in the user preferences of u_0 is exactly the same or the similarity value is above the predefined threshold (0.9).

- SR_2 : The relationship *colleagues* is assigned to the persons depicted in the photo when the photo capture date, time, and location correspond to the preferences of u_0 (namely the stored work days, time, and location).

- SR_3 : When a photo is captured on a weekend day, the rule assigns the relationship *relatives* to a child and an adult depicted in the photo and who appear together in more than one photo within the photo album.

- SR_4 : When two contacts of u_0, each contact on a social network site (SN1 and SN2), have the similarity scores between their name, lastname, and nickname above the predefined thresholds, the rule identifies them as co-referent users.

In this study, we adopted the N3Logic[10] notation since it can be used at the same time for logic and data. Since our model is based on RDF syntax, as shown in Figure 4.8, it is appropriate to use N3Logic. Actually, as stated in the work of Berners-Lee [172], the main goal of N3Logic is to be a minimal extension to the RDF data model such that the same language can be used for logic and data. In addition, N3Logic is a logic that allows rules to be expressed in a web environment.

To extract both of explicit and implicit data from social networks, different functions appropriate to each concept are needed. We list and describe these functions in the next section. In addition, we illustrate within a diagram the data types and functions used in this work.

4.4 Used Functions

In the previous section, we described our data model with its different data types and attributes. The diagram of data types which illustrates the different concepts presented in the previous section is shown in Figure 4.10. In this diagram, we only present the main functions related to different concepts and show their dependency relationships. Furthermore, we provide a representation of the rule model used in this work, as it is illustrated in Figure 4.9.

To allow our framework to function properly, it is important to provide it with the necessary information and values of the used data types. We therefore propose different functions that make use of the data model to extract these information. In the following, we list all the needed functions related to the different data types of our data model. We categorize functions into four

[10]N3Logic, http://www.w3.org/DesignIssues/N3Logic

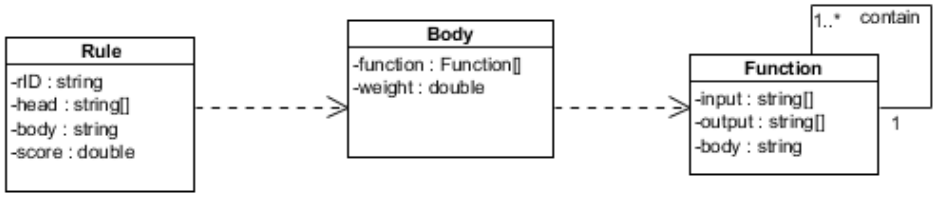


Figure 4.9: Class diagram of our rule model.

categories: 1) functions applied on photos, 2) functions applied on user profiles, 3) functions applied on user preferences, and 4) attributes' functions.

4.4.1 Photo Functions

The following functions are used to extract information from photos. They accept as input one or more photos where a photo is identified by its photo ID (pID_i):

- **date GetPhotoCreationTime():** extracts the time of photo creation and returns it as a date value
- **date GetPhotoCreationDate():** extracts the date of photo creation and returns it as a date value
- **double GetPhotoLocationGPS():** returns the GPS location photo (Longitude/Latitude) of pID_i
- **string[] GetPersonNames():** returns the names of persons depicted in pID_i. This function is applied when persons are already tagged in the photo
- **integer GetNumberPersonsPhoto():** returns the number of persons depicted in pID_i. This function is applied when persons are already tagged in the photo
- **integer DetectFaces():** returns the number of faces detected in pID_i. This function is applied when persons are not tagged. A face detection algorithm detects the face and returns the result
- **integer[] RecognizePersons():** returns the profile identifiers of the persons depicted in the photo by recognizing their faces. This function is applied when persons are not tagged. A face recognition algorithm is needed in this case

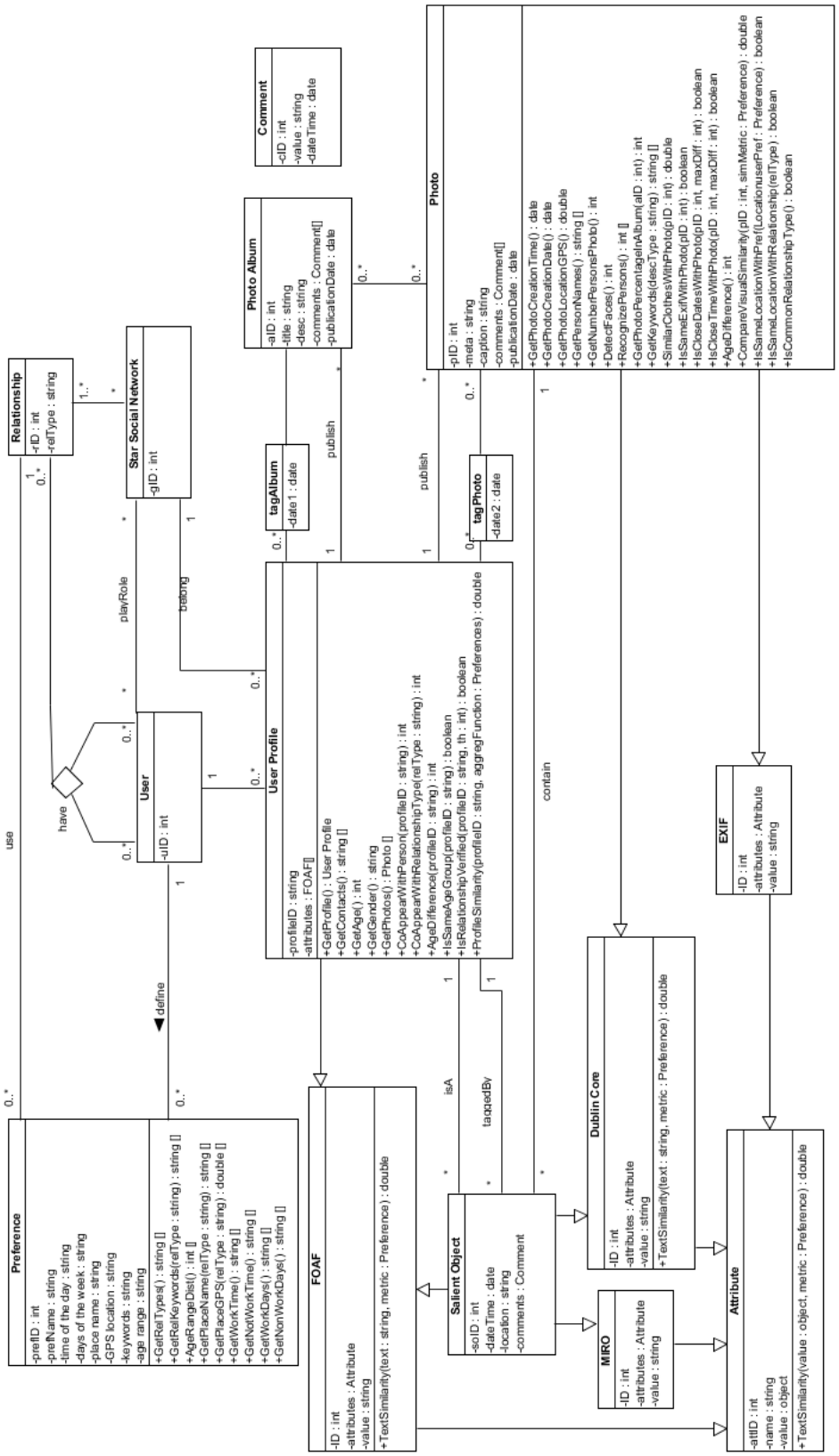


Figure 4.10: Class diagram of our data model.

- **integer GetPhotoPercentageInAlbum(aID: integer):** returns the percentage of photos in the photo album (aID) having their attributes similar to pID_i. If the photo album (aID) is empty, all the photos in the user profile are considered, else only the photos of the defined photo album are considered
- **string[] GetKeywords(descType: string):** returns the keywords of a photo related to the given description type (descType) which can be either the caption, description, or the comments related to a photo
- **double SimilarClothesWithPhoto(pID: integer):** returns a similarity value that indicates how similar are the clothing style between persons depicted within two photos
- **boolean IsSameExifWithPhoto(pID: integer):** compares the Exif attributes (time, date, and location) between two photos. It returns *True* if the two photos have the same Exif attributes, otherwise it returns *False*
- **boolean IsCloseDatesWithPhoto(pID: integer, maxDiff: integer):** returns *True* if the date difference between two photos is less than the maximum date difference (maxDiff), otherwise it returns *False*
- **boolean IsCloseTimeWithPhoto(pID: integer, maxDiff: integer):** returns *True* if the date difference between two photos is less than the maximum time difference (maxDiff), otherwise it returns *False*
- **integer AgeDifference()** returns the difference of age between persons depicted in pID_i
- **double CompareVisualSimilarity(pID: integer, simMetric: Preference):** returns the value that represents the visual similarity between two photos
- **boolean IsSameLocationWithPref(LocationUserPref: Preference)** returns *True* if the place retrieved from the given photo corresponds to one of those indicated in the user preferences, otherwise it returns *False*. The distance per default is the Haversine distance [173], which is a commonly adopted metric for geographical distance
- **boolean IsSameLocationWithRelationship(relType: string)** returns *True* if the place retrieved from the given photo is different than the given relationship type (relType). The *Haversine* distance is the default distance
- **boolean IsCommonRelationshipType()** returns *True* if there is a common relationship type between the persons depicted in pID_i, otherwise it returns *False*

4.4.2 User Profile Functions

In this work, several functions are used to gather information stored within the user profile. A user profile is identified using its profile ID ($profileID_i$). In the following, we list the user profile's related functions:

- **UserProfile GetProfile():** returns the profile of $profileID_i$
- **string[] GetContacts():** returns the list of contacts of $profileID_i$
- **integer GetAge():** returns the age of $profileID_i$
- **string GetGender():** returns the gender of $profileID_i$
- **Photos GetPhotos():** returns all the shared photos and albums of $profileID_i$
- **integer CoAppearWithPerson(profileID: string)** returns the number of times two persons appeared together in the whole set of photos
- **integer CoAppearWithRelType(relType: string)** returns the number of times that a person appeared in photos previously assigned with the given relationship (relType)
- **integer AgeDifference(profileID: string)** returns the difference of age between two persons
- **boolean IsSameAgeGroup(profileID: string)** returns *True* if the two persons belong to the same age group, otherwise it returns *False*
- **boolean IsRelationshipVerified(profileID: string, th: integer)** returns *True* if the confidence of a relationship between two persons is above a given threshold (th), otherwise it returns *False*
- **double ProfileSimilarity(profileID: string, aggregFunction: UserPreferences):** computes an aggregated value of a set of values. Given two profiles (profile1 and profile2), the aim of this function is to return the likelihood that these two profiles refer to the same real-world person. For that the similarity values between the common attributes of both profiles (profile1 and profile2) is computed. Then, we use the Dempster-Shafer theory of evidence (DS) [174] [175], as the default aggregation algorithm

4.4.3 User Preference Functions

Here, we list the different functions used to extract information stored from the "User Preferences". These preferences are related to a main user u_0:

- **string[] GetRelTypes():** returns the list of relationship types used by u_0
- **string[] GetRelKeywords(relType: string):** returns the list of predefined keywords that refer to the given relationship type (relType)
- **string[,] AgeRangeDist(]:** returns the age distribution as defined by u_0. For example, this function returns: (child,13), (teenager, 14-18), (adult, 19-60), and (senior, 61)
- **string[] GetPlaceName(relType: string):** returns the name(s) of the place(s) related to the given relationship type (relType). These names are defined by u_0 (e.g., the name of the enterprize, the street name of the house address, etc.)
- **double[] GetLocationGPS(relType: string):** returns the longitude/latitude related to the given relationship type (relType)
- **string[] GetWorkTime():** returns the working hours of u_0
- **string[] GetNotWorkTime():** returns the non-working hours of u_0
- **string[] GetWorkDays():** returns the weekly working days of u_0
- **string[] GetNonWorkDays():** returns the weekly non-working days of u_0

4.4.4 Attribute Function

In this work, similarity functions can measure the similarity between attributes' values. The similarity function that we use in this work is:

- **double TextSimilarity(text: string, simFunction: UserPreferences):** returns the similarity score between two texts using a given similarity function (simFunction). Choosing an appropriate similarity function must be done carefully with respect to the semantical property of the texts. For FOAF attributes, we propose to use by default the Jaro metric [132] for *Senseless One-term attributes*, the SoftTFIDF metric [176] for *Senseless Multi-terms attributes*, the Explicit Semantic Analysis (ESA) technique [177] for the *Semantic-based attributes*, and the Edit Distance (ED) metric [131] for the *URI and Numeric-based*

attributes. Further details about the metrics and the functions are presented in the next chapters

The functions listed in this section are part of our data model which is described in the previous section and detailed in a class diagram illustrated in Figure 4.10.

4.5 Conclusion

In this chapter we have presented our framework that aims at gathering and analyzing users' profiles and their related information. We also described our data model that incorporates a set of elements available on social networks namely user profiles, photos, photos albums, and salient objects. The gathered information is obtained from these elements by using a set of functions that we provide to extract explicitly stored data or to shape the needed information from existing data. As mentioned before, these functions extract, shape, and infer different information that will be used within semantic rules, by our framework, to enrich links' characteristics among social network users. Clearly comprehending these concepts is deemed a vital part for the understanding of our approach in the following chapters.

In the next chapter, we detail our relationship discovery approach to identify social and co-referent relationship types.

Chapter 5

Relationship Discovery

ABSTRACT

In this chapter, we present our relationship discovery approach. We show the different types of rules (basic and derived) used to identify relationships within the same or across different social networks. We first detail the distinct methodologies to generate the set of basic rules for social and co-referent relationship types. Then, we show how to extend the basic rules in order to obtain the set of derived rules. These derived rules take into account the context of each user and are interestingly useful to identify relationship types. Finally, we present our relationship discovery algorithm.

5.1 Introduction

As mentioned before, social relationships between users and their contacts are rarely explicitly exhibited on social networks. In this chapter, we aim to discover non-identified social relationships (e.g., colleagues, relatives, friends, etc.). To deal with this issue, we propose a rule-based approach able to exploit several characteristics and features of social networks data, in particular, users' profiles and shared photos. Our approach is able to identify appropriate rules independently of the relationship types to be discovered. These rules are called intra-social rules since they target social relationships between a main user and her contacts within a single social network. In this study, we distinguish between two types of rules: 1) basic rules, 2) derived rules. We therefore propose a methodology to automatically generate basic rules that correspond to the social relationship types to be discovered. In addition, we manually created a set of common sense rules to enhance the results achieved by the set of basic rules. Common sense rules are independent of the relationship types to be discovered.

In addition, we present a methodology that can generate inter-social basic rules. Inter-social rules aim to identify co-referent users among the contacts of main users across different social networks. In fact, co-referent basic rules depend on the studied social networks. Our approach consists in exploiting all the profiles' attributes of the users which include different types of data from explicit to implicit information and from textual to multimedia information. We describe two ways to generate basic rules that can be incorporated in our work: attribute-based basic rules and profile-based basic rules. Both of the rules are automatically generated by our approach. Moreover, similarity metrics and decision making functions are discussed in this chapter since they are included in our methodology.

Finally, in this chapter, we aim to enhance relationship discovery results. To this end, we suggest to use the set of basic rules to generate a new set of rules called derived rules Derived rules are generated using mining techniques which successfully adapt the derived rules to the context of each user. We explain the method to generate the set of derived rules. We also describe the relationship discovery algorithm that underlines the use of the basic and derived rules to identify social and co-referent relationships between social network users.

The rest of this chapter is organized as follows. In Section 5.2, we present the rules for social relationship discovery and describe our methodology to generate the set of social basic rules. Then, in Section 5.3, we present the rules for co-referent relationship discovery along their associated generation methodology. In Section 5.4, we present the set of derived rules and how to generate them. Finally, in Section 5.5, we show our relationship discovery algorithm where

basic and derived rules are used to identify relationships among users.

5.2 Rules for Social Relationship Discovery

On social network sites, users tend to shape their online contacts in the same way as their offline or real-life contacts. For instance, a user can be connected to different types of contacts such as colleagues, relatives, friends, etc. However, within current social network sites, the types of social relationships are not explicitly available [3] [4]. It is therefore of particular interest that a relationship discovery approach detects the type of relationships between social network users. This would certainly contribute positively to the organization of users' contact management, facilitate its personalization, help enhance the level of personal privacy protection, and its simplification as an action toward a better profile management. Nevertheless, predicting the different types of relationships in social network sites has not been properly explored since:

- Several networking sites provide users with exclusive relationship type. It is common that social network users indicate other persons as friends even though they do not particularly know or trust them [11]. Treating all online contacts in the same way, without differentiating one contact from another, is an unsafe and restrictive practice since users may need to share some data (notes, photos, videos, etc.) [178] and interact with some of their contacts while preserving their relationship privacy [179]. Indeed, users quit such networking sites when their bosses, friends, and family are all presented with the same online persona [178].

- Some social networking sites provide the possibility to define manually how a user knows each of her contacts. However, most of the time, this option is skipped by social network users and only the link existence is indicated [7]. Thus, several relationship types remain unlabeled.

- On social network sites, the provided communication tools (e.g., messages, posts, photo sharing, etc.) cannot reveal the relationship types among users. In fact, frequent interactions using these communication tools are not indicative of relationship types since these tools can be used regularly to communicate for various purposes with different types of contacts (e.g., colleagues, relatives, friends, etc.). In addition, interaction factors (e.g., shared photos, status updates, comments, etc.) based on these tools, which are able to identify the type of relationships' closeness, are yet missing [180]. Consequently, the intensity of interaction through existing communication tools cannot predict the type of relationship (e.g., personal or professional).

- Users' relationships dynamically evolve and change over time. A friend becomes a family member, a colleague becomes a friend, a friend becomes a colleague, etc. Also, a friend can be a colleague and a family member at the same time. Updating relationships that a user has with her contacts requires constant maintenance which is a tedious and time-consuming duty [5]. Taking into consideration the flexible nature of social networks, a user must be able to potentially deal with every modification of her contact list, and every relationship type variation with a contact.

For all these reasons, we believe that social network users need support to predict (at least semi-) automatically social relationship types that they share with their contacts within a single social network site. This work is specifically concerned with the task of link type prediction, and from now on we will use "link type prediction" and "link type discovery" interchangeably to refer to this task.

Before giving a brief overview of our methodology, we discuss the motivation behind our choice of using in this work the following social relationship types: colleagues, relatives, and friends.

5.2.1 Main Relationship Types

As mentioned previously, we focus on three categories that represent the most popular types of relationships within current social network sites. Based on the findings of an interesting survey conducted by Pew[1] statistics, we decided to group the relationship types as follows: colleagues, relatives, and friends.

5.2.1.1 Colleagues

The survey above-mentioned reveals that social network users have met 22% of their contacts at high school, while only 9% at college. In addition to these two categories, 10% of the contacts are the co-workers of social network users. We decided to group high school contacts, college contacts, and co-workers in one category called *colleagues*. In general, these different sets of contacts refer to persons that social network users usually meet within the context of professional activities including studies and work. The total percentage of colleagues is 41% of all social network contacts.

[1] http://www.pewinternet.org/Press-Releases/2011/Social-networking-sites-and-our-lives.aspx

5.2.1.2 Relatives

We refer to *relatives* as members of either the immediate family (father, mother, kids, sister, brother) or extended family (grandfather, grandmother, aunt, uncle, cousin, etc.) The overall percentage of the relatives is 20% of the total percentage of the contacts, 8% are from the immediate family and 12% are from the extended family.

5.2.1.3 Friends

Remaining contacts are categorized as random friends that represent 31% of all social network contacts. Contacts in this category are persons that social network users met within different real life events, we refer to them as *friends*.

It is important to note that two other categories exist: 1) 7% as persons from voluntary groups, and 2) 2% as social network users' neighbors. The low percentage of these two categories may be attributable to the fact that, even today, not everyone is friend with their neighbors or participates in voluntary groups. Hence, we don't consider these two relationship types and we only focus in this work on the three main relationship types (colleagues, relatives, and friends).

5.2.2 Methodology Overview

We propose a rule-based methodology as a way to discover the type of social relationships that a user has with her contacts. The main originality of our approach consists in generating automatically and extending appropriately rules for each relationship type to identify. In fact, automatically generating rules can significantly enhance the relationship discovery process since 1) it relieves social network users from the complex task of manually creating rules, and 2) it can generate a wide variety of rules that a user may not be able to identify manually.

For social relationship type discovery, we generate the set of basic rules, all of which considered as being the most trustworthy. In fact, these rules are based on the wisdom of a big number of social network users. The basic rules generation methodology is detailed in Section 5.2.3. In addition, we propose another set of rules named common sense rules. Actually, this set of rules is based on our common sense. The aim behind creating common sense rules is to identify further social relationship types. These common sense rules are independent of relationship types to be discovered. They can be used without applying basic rules whenever a user has some social relationship types already identified within her social network. Note that our methodology takes into account the worst case scenario where all relationship types are initially unlabeled. In such

case, the set of basic rules is firstly applied to identify some relationship types before using the set of common sense rules. Common sense rules are listed in Section 5.2.3.3.

It is to be recalled that within the context of the same social network site, we refer to the basic and common sense rules as *intra-social rules* since they target links between a user and her contacts. The next section describes the methodology that we propose to generate the set of intra-social basic rules.

5.2.3 Basic Rules Generation Methodology

The main goal here is to automatically provide the set of basic rules for discovering social relationship types between a main user and her contacts, particularly using the wisdom of the crowd. Here, we aim to achieve two specific targets as illustrated in Figure 5.1:

1. First of all, we intend to automatically construct a collection of photos for each relationship type to be discovered (such as colleagues, relatives, friends, etc.).
2. Second, we aim to obtain the set of basic rules to be generated from the constructed collection of photos. In fact, each photo collection contains photos that capture real-life situations related to each relationship type to discover.

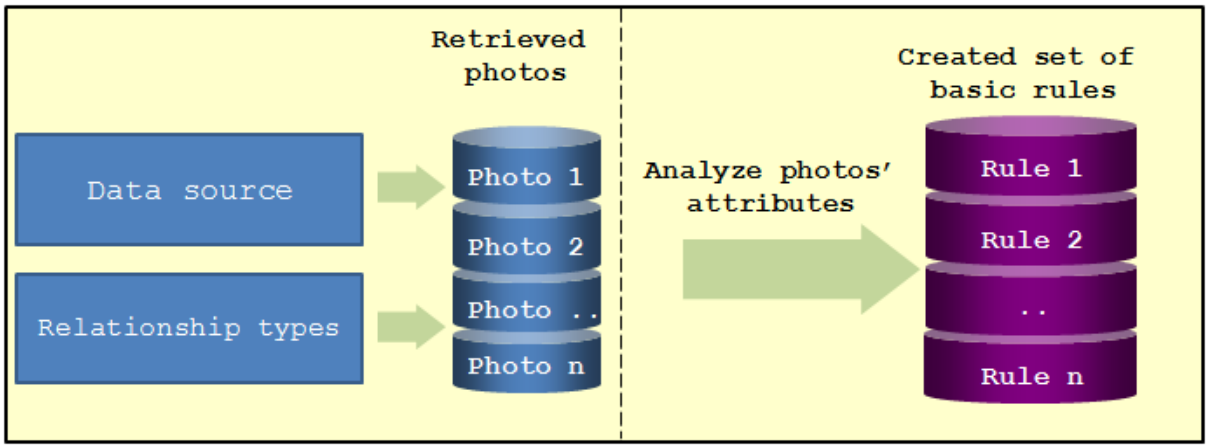


Figure 5.1: Our basic rules generation methodology composed of two parts: 1) constructing the photo dataset, and 2) generating the set of basic rules.

In the following, we start by describing the different steps required to build our photo dataset, taking into account the previously mentioned requirements, prior to generating the set of basic rules.

5.2.3.1 Building the Photo Dataset

With the astounding rise of social network sites (such as Facebook, MySpace, LinkedIn, etc.) and content-sharing sites with social networking functionalities (such as Youtube for sharing videos, Flickr for sharing photos, etc.), a growing interest in the automatic construction of datasets (e.g., user profiles, photos, tweets, etc.) is motivating many approaches in different fields (e.g., active learning [181], computer vision [182], etc.).

Hereunder, we define the requirements and steps to build a photo dataset. These steps involve capturing collective knowledge, determining representative social relationships, verifying the availability of metadata, and preprocessing the photo dataset.

Photo Dataset Requirements

Despite the wide interest in photos and the wealth of photos published on content-sharing sites, automatically constructing photo datasets to build rules remains a challenging endeavor [183] [184]. In this study, we identified the following requirements that should be taken into account when constructing a photo dataset to represent different social relationship types:

- **Representative of any social relationship type:** each social relationship type must be illustrated correctly with a set of gathered photos semantically related to a specific relationship type. Actually, many different relationship types between persons naturally exist (e.g., parent, sister, best friend, co-worker, etc.), and the dataset must be capable to represent these relationships.

- **Available metadata:** metadata information must be embedded within each photo available in the dataset. Of particular importance among various embedded metadata in photos are the date, time, and GPS location. Note that previous works in the field of relationship discovery did not consider the use of these metadata information [167] [92]. In practice, only a small number of photos, within publicly available photo datasets (such as the MIR Flickr dataset [185]), includes all of these three metadata information.

- **Photos depicting persons:** photos can depict a range of subjects, from animals and natural scenes, to documents, persons, and more. Consequently, when the goal is to gather photos of persons, one must verify that at least one person is depicted within each collected photo.

Note that the above-mentioned requirements are included in each step of constructing the photo dataset.

Capturing Collective Knowledge through Crowdsourcing

The collected dataset must reflect the point of view of a large number of users with diverse cultural backgrounds and areas of expertise. In fact, it is clear that a large number of users helps ensure a more reliable and accurate labeling/annotation. Furthermore, asking experts to annotate photos with labels/annotations related to one (or several) relationship type is not practical since it is an expensive and time-consuming process. An alternative to using experts' annotation is crowdsourcing [186] that harnesses collective knowledge to assign photos with semantically corresponding labels. This is a reasonable alternative to obtain quickly and without much efforts an already annotated collection of photos for any social relationship type. In fact, crowdsourcing is a practice that takes advantage of the "wisdom of the crowd" [187]. Though photos annotated by non experts may seem unreliable to use, the collective wisdom of the crowd can guarantee good annotation results by leveraging large-scale web communities. Different types of applications use crowdsourcing for labeling large datasets [188], discovering region-based landmarks [189], and constructing photo collections [185], etc. By outsourcing a task to an internet-scale community, crowdsourcing can reveal interesting and non-overlapping information [190]. As a result, it is possible to obtain photos already annotated by a large number of web users with keywords related to different relationship types.

As mentioned earlier, content-sharing sites (such as Flickr) and social networking sites (such as Facebook) are valuable sources of annotated photos. Photo tagging has gained success and attention within content-sharing and social network sites. Its success contributes to a number of benefits that provide different services and applications, namely: 1) photo searching, 2) photo browsing, and 3) person identification. In addition to these three services, which are commonly used by social network users, we propose to exploit available labels to construct relationship-specific datasets. As explained in the following, building the photo dataset starts by querying publicly available photos. Photos with relevant labels related to a specified relationship type (chosen by the main user) are selected. For example, for the *relatives* relationship type, only photos tagged with keywords such as "family", "father", "mother", etc. are downloaded to build the *relatives* photo dataset.

Determining Representative Social Relationships

Whether by choice or circumstance, social network users tend to connect to different types of contacts. This is why we believe that it is important to give users the possibility to choose the relationship types they wish to discover. However, with no control mechanism over the chosen relationship type, results may suffer mainly from two problems:

- **Relationship type relevancy:** a user may choose to identify a relationship type that is not related to any real-life relationship, i.e. a word that semantically doesn't correspond to a relationship type (e.g., car, mountain, cedar, etc.). Consequently, verifying if a word is conform with a real-life relationship comes down to checking whether the chosen word is part of the *Relationship* vocabulary[2], which includes a variety of concepts to describe relationships between persons (such as *Colleague Of*, *Works With*, *Parent Of*, *Sibling Of*, *Friend Of*, *Acquaintance Of*, etc.). Indeed, whenever a user chooses a relationship type to discover, the system checks the relevancy of the proposed word against the set of specified concepts in the *Relationship* vocabulary. This chosen word is used as a relationship type only if it exists (or one of its semantically similar word(s) exists) within the *Relationship* vocabulary. Otherwise, the main user is asked to choose another word. By using a lexical dictionary, for instance Word-Net [191], the chosen word can be semantically related (e.g., Synonym, Hyponym, Hypernym, Meronym or Holonym) to one of the concepts available in the *Relationship* vocabulary. For instance, chosen words such as "relatives" or "family" are related to the concept "parent", the chosen word "associate" is related to the concept "colleague", etc.

- **Data availability:** Data is an essential asset for building the photo dataset and thus for creating the set of basic rules. Since our basic rule generation methodology is based on real-world data, it is of great importance to collect a representative set of photos that correspond to the chosen relationship type. Consequently, the number of photos, extracted from the photo sharing site, must be above a specified threshold. In this study, we set this threshold to be equal to 300 photos per relationship type to discover. When there are not enough photos related to a chosen word (less than 300 photos), the system proposes to the main user another semantically similar word to use.

Verifying the Availability of Metadata

As the number of shared photos continues to grow, the number of photos embedded with metadata information will likely continue to increase. Metadata information that are of particular interest in this work include the capture date and the capture time of the photo, along with the GPS location information. The availability of these metadata allows, to a varying degree, the possibility to assign each relationship type with its most frequently corresponding metadata values (e.g., photos taken within work time (time metadata) and on a weekeday (date metadata) are most probably assigned to "colleagues" relationship type). Thus, making use of metadata would help identify metadata values of specific relationship types accurately and reliably, and

[2]http://vocab.org/relationship/.html

that could be used to discriminately characterize a relationship type. However, most of the time, metadata values are not discriminatory factors alone, but instead they must be associated with other explicit and implicit information (c.f. Chapter 4) to identify a relationship type. In this work, we verify the availability of these metadata information before considering the photo as valid.

Preprocessing the photo dataset

Given the collected photos, we implemented a face detection algorithm [192] to verify whether or not there is any face in the image. However, using a face detection algorithm doesn't ensure a proper face detection mainly due to the fact that: some objects can be detected as faces but in reality they are not, and 2) some faces in the images are not detected by the algorithm. To solve these issues, we built an annotation tool to fix such problems. This tool allows to categorize manually the gender of these persons (male or female), and their age category (baby, child, adult, and senior) since we did not adopt any image processing technique for gender and age estimation.

In the following, we present the methodology to generate automatically the set of basic rules.

5.2.3.2 Generating the Basic Rules

In order to automatically generate basic rules, from the photo dataset we extract information that we represent as a set of rules depending on the social relationship types. This yields relationship-specific information, obtained from the already built photo dataset, based on the user-defined relationships' choice. In fact, extracted information is processed by the functions listed in Chapter 4, and thus this brings interesting and non-trivial information that is transformed into rules for each relationship type. Algorithm 1 summarizes the process of generating the set of basic rules from the photo dataset.

Identifying Useful Information

For the identification of useful information part of a rule's body, practically all functions listed in Chapter 4 come to play. A key challenge for rule generation is to identify these attributes; simply using explicit information (e.g., embedded metatada, tags, descriptions, etc.) might not perform as well as implicit information (number of co-appearance, age group category, etc.) that requires additional efforts to define and infer corresponding information.

Indeed, there are several possible explicit and implicit information that can be used. We identified some of the most important ones for this work. However, the choice of a main user is

not only limited to these chosen information (e.g., a main user can choose to use different Exif and metadata information). Actually, in this work we used:

1. **Explicit information:**
   (a) the metadata of the photo, namely the photo's date, time, and GPS location,
   (b) the description and the tag of the photo.
2. **Implicit information:**
   (a) the photo cardinality (number of persons depicted in the photo),
   (b) the age difference between persons depicted in the photo,
   (c) the gender of the persons depicted in the photo,
   (d) the number of occurrences of each person within a photo album.

Representing Extracted Information using Rules

Essentially, information available within photos can be classified into explicit information (stored data) and implicit information (inferred data). These two specific kinds of information store an important amount of data, which, if extracted and utilized properly, can be converted into valuable knowledge and hence into rules representing this knowledge.

Before detailing the attributes that we use in this work, it is important to note that a rule, as introduced in Chapter 4, is represented as follows:

$$R:\ body \longrightarrow \langle head, score \rangle$$

Information extracted from each photo is written in the form of rules where the body of the rule is composed of a conjunction of conditions, and the head of the rule holds the social relationship type to which a given photo belongs within the dataset. When all the conditions in the body of the rule are satisfied, then the social relationship type, which is the value of the rule's head, is assigned to the relationship that links a main user to one or several of her contacts.

After transforming the knowledge to rules, we may obtain a big number of rules that describe each social relationship type chosen by the main user. Inevitably, a varying number of noisy rules can be also generated and consequently must be eliminated since they cannot characterize any relationship type. To address this issue, the low frequency rules of each relationship type must be discarded. Given an arbitrary rule, its support score is measured. The support score is the percentage of the relative occurrences of rules satisfying the body and the head of this given

rule within the overall set of rules. Rules having a support score below a specified threshold are discarded. We set this threshold to be equal to 10%. Rules having a threshold above the specified one represent the set of basic rules.

Algorithm 1: **Generate Basic Rules**

Input:
photoDataset[], // The constructed photo dataset based on the
// user-defined relationship types
1 informationToExtract[] // The set of actual attributes to extract
Data:
supportThreshold // The manually predefined support threshold
Output:
basicRules[] // The set of basic rules
2 **begin**
// The attributes of each photo available in the photo dataset are extracted to build
// the basic rules. For each photo corresponds a rule.
3 **foreach** $photo_i$ *in photoDataset* **do**
// Get the relationship type assigned to $photo_i$ from the photoDataset
4 photoRelationshipType ⟵ photoDataset[$photo_i$].RelationshipType ;
5 **foreach** $information_j$ *in informationToExtract* **do**
// Extract the value of $information_j$ from $photo_i$
// Store the value of each attribute within the body of a specific $BasicRule_i$
// along with a relationship type value for the head
6 basicRules[] ⟵ ($photo_i$,$information_j$,photoRelationshipType) ;
7 **end**
8 **end**
// Discard rules with low support score
9 **foreach** $basicRule_i$ *in basicRules* **do**
// Compute the support score (supportScore) of $basicrule_i$
10 supportScore ⟵ ComputeSupportScore($basicrule_i$) ;
// The default value of the supportThreshold is 10%
11 **if** *supportScore* < *supportThreshold* **then**
12 basicRules.Delete($basicRule_i$);
13 **end**
14 **end**
15 Return basicRules[];
16 **end**

5.2.3.3 Common Sense Rules

Based on common sense, we generated a set of rules to help enhance and improve the previous set of basic rules. In this study, common sense rules are invoked only after the application of

the set of basic rules. The common sense rules that we designed are independent from the social relationship types and can be used to discover different types of contacts. Note that a significant number of photos must be available and satisfy the conditions of a common sense rule so that the rule can be applied. We set this percentage to be equal to 25% of all the photos of a main user. Actually, new relationship types can be detected and new rules can be created from previously discovered relationship types. In fact, common sense rules can:

1. **Identify a relationship type when:**

   (a) **A contact that appears with other persons of an already identified relationship:** when a contact (with unlabeled relationship type) appears with several persons having a labeled relationship, then we can infer that this contact is assigned with the same type of relationship. For example, if 3 persons out of 4 are already known as "colleagues", then we can infer that the 4th person is a colleague as well.

   (b) **A contact that appears only with persons of a single relationship type:** when a contact has a significant number of photos only with persons of a single social relationship type, then we can infer that she similarly shares the same relationship type with the main user. For example, a contact who appears in a significant number of photos with family members is considered to be a relative as well.

   (c) **Photos have the same Exif information:** when a photo, $photo_1$, has its Exif attributes (date, time, and location) similar to a significant set of photos $photoset_a$ of an already identified relationship, then we can infer that the persons who appear in $photo_1$ are also assigned the same relationship identified in $photoset_a$.

2. **Create a new rule when:**

   (a) **Contacts with already identified relationships:** when persons depicted in $photo_1$ have their relationship type, R_1, already identified via other photos, however the attributes of a $photo_1$ don't match with any basic rule, then we can infer that the attributes of $photo_1$ can also be used to generate a new basic rule that identifies the relationship type R_1.

   (b) **Same clothing:** persons usually do not change their clothing when they appear in a set of photos during a short period of time (e.g., a given day). If the clothing of the persons in $photo_1$ is similar to the clothing of the persons in the set $photoset_a$ of identified relationship R_1, then we can infer that these persons are the same and therefore we can use the attributes of $photo_1$ to generate a new basic rule that identifies the relationship type R_1.

These common sense rules help enhance and improve the results returned by the set of basic rules as it will be demonstrated in our experimentations in Chapter 7.

In the next section, we present the set of basic rules that we obtained by applying the basic rules generation methodology.

5.2.4 Generated Basic Rules

In this section, we detail the obtained set of basic rules that aim to discover the social relationship types that link a main user to her contacts. These basic rules are considered as being trustworthy with respect to their corresponding relationship types.

Using our proposed methodology to build a photo dataset (as described earlier), we gathered 1200 photos for three relationship types: colleagues, relatives, and friends (400 photos for each relationship type), and generated the set of basic rules related to an adult main user. In fact, according to a study about social network sites published by Pew[3] statistics, 74% of social network users in 2010 were adults (age group 18-50 years). Therefore, in this work we focus on this large age group and consider the case where the main user is an adult.

As stated earlier, basic rules are generated automatically using our methodology. These rules are classified into three categories "Colleagues-related rules", "Relatives-related rules", and "Friends-related rules". These rules are *ordered*, this means that first we apply the general rules, then rules to discover colleagues, followed by rules to discover relatives, and finally rules to discover friends. However, among these three categories of rules, we identified some rules having the same rule's body and different heads. We refer to such rules as "General rules". These rules are common regardless of the social relationship to be discovered. They associate a photo to a social relationship type if the photo's keywords are similar to the ones stored in the user preferences or if the photo's location refers to a place's location which is also stored in the user preferences. In this work, we provided the system with a set of predefined keywords that describe the relationship types to be discovered. Table 5.1 presents the keywords that we believe correspond to each relationship type.

In the following, we show the most relevant basic rules that we obtained. Note that these basic rules have a score equal to 1.

[3]http://www.pewinternet.org/Reports/2009/Generations-Online-in-2009.aspx

Social Relationship Type	Keywords
Colleagues	Employee, Associate, Co-worker, Boss, Vendor, Customer, Client, Work, Desk
Relatives	Family, Relative, Brother, Sister, Cousin, Daughter, Son, Granddaughter, Grandson, Grandparent, Aunt, Uncle, Father, Mother, Spouse, Niece, Nephew
Friends	Friend, Girl/Boy Friend, Best Friend, Acquaintance

Table 5.1: Relationship Keywords

5.2.4.1 General Rules

These rules are applied to all photos, and are remarkably useful to identify the relationship type in photos where only one person is depicted. In fact, photo's keywords and metadata information can be used to discover the type of social relationship in different situations, in particular when only one person is depicted in a photo.

In Algorithm 2, we compute the similarity between the keywords extracted from the photo (which is a combination of the description, caption, place name, and comments) and the keywords stored in the user preferences for each relationship type. Similarly, the GPS location of the photo is compared to the GPS locations of each relationship type. If the similarity score between the keywords is above a specified threshold or the photo location refers to a location associated to a relationship type, then the person(s) depicted in the photo is/are assigned with the corresponding relationship type.

5.2.4.2 Colleagues-related Rules

These rules state that persons depicted in a photo are assigned with a "colleagues" relationship type. The following rules, summarized in Table 5.3, are applied on photos where two or more persons are depicted:

- Rule 1: when two adults of the same gender appear together in one photo, and if the photo capture day is a weekday and the time is within working hours.
- Rule 2: when two adults of different genders appear together in one photo, if the photo capture time is within working hours.
- Rule 3: when a group of adults appear together in one photo, if the photo capture day is a weekday and the time is within working hours.

Algorithm 2: **General Basic Rules**

```
   Input:
   u_0, // The profile of a main user
1    basicRules[], // The set of basic rules
2    photosProfiles[] // The set of actual photos and profiles to use with the rules
   Data:
   threshold // The manually predefined similarity threshold
3  begin
4      foreach photo_i in photoDataset do
5          foreach basicRule_i in basicRules do
6              (basicrule_i is valid) WHEN
7              KeywordsPhoto = photo_i.GetKeywords("description") &
               photo_i.GetKeywords("caption") & photo_i.GetKeywords("comments");
8              locationPhoto = photo_i.GetPhotoLocationGPS();
9              foreach relType in u_0.GetRelTypes() do
10                 keywordsUserPref = u_0.GetRelKeywords(relType) &
                   u_0.GetPlaceName(relType) ;
11                 locationUserPref = u_0.GetLocationGPS(relType) ;
12                 if (keywordsPhoto.TextSim_metric(keywordsUserPref) ≥ threshold) OR
                   (photo_i.IsSameLocationWithPref(locationUserPref)) then
13                     photo_iGetPersonNames() ⟵ relType ;
14                 end
15             end
16         end
17     end
18 end
```

- Rule 4: when a group of adults appear together in one photo, if the photo capture day is a weekend and the time is out of the working hours.

5.2.4.3 Relatives-related Rules

The following rules state that persons depicted in a photo are assigned with a "relatives" relationship type. All the following rules are applied to photos taken in a different place than the work place. These rules, summarized in Table 5.3 and Table 5.4, are detailed below:

- Rule 1: when two persons, one of them is a baby and the other is a female adult, appear in a significant number of photos, and if the capture day of the photos is a weekend.
- Rule 2: when two persons, one of them is a baby and the other is a male adult, appear in one photo, if the photo capture day is a weekend and the time is out of the working hours.

Table 5.2: Basic rules of the relationship colleagues written in N3Logic format.

Rule 1: @forAll X, Y. {X appears_in p_1. Y appears_in p_1. X, Y Adults, Males. p_1 time:day weekday. p_1 time:hour workHours. } log:implies {X colleague Y} /1 @forAll X, Y. {X appears_in p_1. Y appears_in p_1. X, Y Adults, Females. p_1 time:day weekday. p_1 time:hour workHours. } log:implies {X colleague Y} /1
Rule 2: @forAll X, Y. {X appears_in p_1. Y appears_in p_1. X, Y Adults. X a Male. Y a Female. p_1 time:hour workHours. } log:implies {X colleague Y} /1
Rule 3:@forAll X_1,...,X_n. {X_1,...,X_n appear_in p_1. X_1,...,X_n Adults. p_1 time:day weekday. } log:implies {X_1,...,X_n colleagues} /1
Rule 4: @forAll X_1,...,X_n. {X_1,...,X_n appear_in p_1. X_1,...,X_n Adults. p_1 time:day weekend. p_1 time:hour nonWorkHours. } log:implies {X_1,...,X_n colleagues} /1

- Rule 3: when two persons, one of them is a baby and the other is a senior, appear in few photos, if the capture day of the photos is a weekday and the time is out of the working hours.
- Rule 4: when two persons, one of them is a teenager and the other is an adult, appear in a significant number of photos, if the capture day of the photos is a weekday and the time is out of the working hours.
- Rule 5: when two persons of different gender, one of them is a teenager and the other is a senior, appear in one photo, and if the capture day of the photo is a weekend and the time is within working hours.
- Rule 6: when two adults of different gender appear in a significant number of photos, if the capture day of the photos is a weekend and the time is within working hours.
- Rule 7: when two babies and two adults appear together in a significant number of photos,

if one of the adults is a male and the other is a female, and if the capture day of the photos is a weekend and the time is within the working hours.

- Rule 8: when two adults and one baby appear together in a significant number of photos, if one adult is male and the other is female, and if the capture day of the photos capture is a weekday.
- Rule 9: when two adults with a baby and a teenager, appear together in at least one photo, if the capture day of the photo is a weekend and the time is within working hours.
- Rule 10: when a group of adults and one senior person appear together in one photo, if the capture day of the photos is a weekend and the time is out of the working hours.
- Rule 11: when a group of teenagers appear with two adults (one male and one female) in few photos, if the capture day of the photos is a weekend and the time is within working hours.

5.2.4.4 Friends-related Rules

The following rules state that persons depicted in a photo are assigned with a "friends" relationship type. All these rules refer to photos taken in a different place than the workplace. These rules apply on persons who don't appear with the family members of the main user.

These rules, summarized in Table 5.4, are listed below:

- Rule 1: when two teenagers appear together in a significant number of photos, if the capture day of the photos is a weekend and the time is within working hours.
- Rule 2: when two adults of the same gender appear in a significant number of photos, and if the capture day of the photos is a weekend.
- Rule 3: when two adults appear in a significant number of photos, if the capture day of the photos is a weekday and the time is out of the working hours.
- Rule 4: when a group of adults appear together in few photos, if the capture day of the photos is a weekday and the time is out of the working hours.
- Rule 5: when a group of male and female adults appear together in a significant number of photos, and if the capture day of the photos is a weekend.

Table 5.3: Basic rules of the relationship relatives written in N3Logic format (Part 1).

Rule 1:@forAll X, Y. {X, Y appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X a Baby.
Y an Adult, a Female.
p_1 time:day weekend. }
log:implies {X relative Y}/1
Rule 2:@forAll X, Y. { X, Y appear_together_in p_1.
X a Baby.
Y an Adult, a Male.
p_1 time:day weekend.
p_1 time:hour nonWorkHours. }
log:implies {X relative Y}/1
Rule 3: @forAll X, Y. { X, Y appear_together_in p_1,...,p_n.
n math:lessThan significant_nb.
X a Baby.
Y a Senior.
p_1 time:day weekday.
p_1 time:hour nonWorkHours. }
log:implies {X relative Y}/1
Rule 4:@forAll X, Y. {X,Y appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X a Teenager.
Y an Adult.
p_1 time:day weekday.
p_1 time:hour nonWorkHours. }
log:implies {X relative Y}/1
Rule 5: @forAll X, Y. {X, Y appear_together_in p_1.
X a Teenager, a Male.
Y a Senior, a Female.
p_1 time:day weekend.
p_1 time:hour workHours. }
log:implies {X relative Y}/1

@forAll X, Y. {X, Y appear_together_in p_1.
X a Teenager, a Female.
Y a Senior, a Male.
p_1 time:day weekend.
p_1 time:hour workHours. }
log:implies {X relative Y}/1
Rule 6:
@forAll X, Y. { X, Y appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X an Adult, a Male.
Y an Adult, a Female.
p_1 time:day weekend.
p_1 time:hour workHours. }
log:implies {X relative Y}/1

Table 5.4: Basic rules of the relationship relatives written in N3Logic format (Part 2).

Rule 7:
@forAll X, Y, Z, W. { X, Y, Z, W appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X an Adult, a Male.
Y an Adult, a Female.
Z, W Babies.
p_1 time:day weekend.
p_1 time:hour workHours. }
log:implies {X, Y, Z, W relatives}/1
Rule 8:
@forAll X, Y, Z. { X, Y, Z appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X an Adult, a Male.
Y an Adult, a Female.
Z Baby.
p_1 time:day weekday.
log:implies {X, Y, Z relatives}/1
Rule 9:@forAll X, Y, Z, W. { X, Y, Z, W appear_together_in p_1,...,p_n.
n log:greaterThan 0.
X, Y Adults.
Z a Baby.
W a Teenager.
p_1 time:day weekend.
p_1 time:hour workHours. }
log:implies {X, Y, Z, W relatives}/1
Rule 10:@forAll X_1,...,X_n, Y. { X_1,...,X_n, Y appear_together_in p_n.
n math:equalTo significant_nb.
X_1,...,X_n Adults.
Y a Senior.
p_1 time:day weekend.
p_1 time:hour nonWorkHours. }
log:implies { X_1,...,X_n, Y relatives}/1
Rule 11:
@forAll X_1,...,X_n, Y, Z. { X_1,...,X_n, Y, Z appear_together_in p_1. n math:greaterThan 2.
X_1,...,X_n Teenagers.
Y an Adult, a Male.
Z an Adult, a Female.
p_1 time:day weekend.
p_1 time:hour workHours. }
log:implies { X_1,...,X_n, Y, Z relatives}/1

Table 5.5: Basic rules of the relationship friends written in N3Logic format.

Rule 1: @forAll X,Y. { X,Y appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X,Y Teenagers.
p_1,...,p_n time:day weekend.
p_1,...,p_n time:hour workHours. }
log:implies {X friend Y}/1
Rule 2:@forAll X,Y. { X,Y appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X,Y Adults, Males.
p_1,...,p_n time:day weekend. }
log:implies {X friend Y}/1

@forAll X,Y. { X,Y appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X,Y Adults, Females.
p_1,...,p_n time:day weekend. }
log:implies {X friend Y}/1
Rule 3:@forAll X,Y. { X,Y appear_together_in p_1,...,p_n.
n log:equalTo significant_nb.
X,Y Adults.
p_1,...,p_n time:day weekday.
p_1,...,p_n time:hour nonWorkHours. }
log:implies {X friend Y}/1
Rule 4:@forAll X_1,...,X_n. { X_1,...,X_n appear_together_in p_1,...,p_n.
n math:lessThan significant_nb.
X_1,...,X_n Adults.
p_1,...,p_n time:day weekday.
p_1,...,p_n time:hour nonWorkHours. }
log:implies { X_1,...,X_n friends}
Rule 5:@forAll X_1,...,X_n. { X_1,...,X_n appear_ together_in p_1,...,p_n.
n math:equalTo significant_nb.
X_1,...,X_n Adults.
p_1,...,p_n time:day weekend. }
log:implies { X_1,...,X_n friends}/1

5.2.5 Discussion

In this section, we presented our approach to deal with the problem of relationship type discovery. In fact, we address this problem by leveraging the characteristics and features of social networks, namely by using, on one hand, the available information within users' preferences/profiles and, on the other hand, the embedded information extracted from shared photos in which users' contacts are depicted so to measure and qualify the types of relationships between users.

Our approach goes beyond analyzing traditional communication data and classification. Constructing for a user, different social graphs of relationships where the connection links are enriched by their types, is important for relationship management. This is particularly important since it enables users to visualize each graph of contacts grouped by their relationship type.

Our contributions consist of:

1. Providing an automatic way to build a photo dataset where photos are retrieved from a content-sharing social site (such as Flickr). We identified different criteria to be respected in order to use this photo dataset in our proposed basic rule generation methodology.
2. Describing a methodology to generate the set of basic rules after building a photo dataset. This methodology is able to generate the set of basic rules related to different relationship types according to the user's choice.
3. Devising a set of common sense rules that are independent from the types of relationships to be discovered. The aim of these common sense rules is to enhance the results returned by the set of basic rules.
4. Exploiting in details explicit and implicit data by making use of embedded metadata information (date, time, and GPS) and photos' tags, in addition to other implicit data that we extract by analyzing photos (persons' co-appearing together, the age and gender category of the depicted persons, etc.).

In the next section, we will introduce our proposal to generate co-referent rules across different social network sites.

5.3 Rules for Co-referent Relationship Discovery

On the web, social networks are highly dynamic [70]. They offer popular and rich platforms for diverse online social activities. Among these activities, one can cite collaborative tagging

(e.g., Flicker[4]), blogging sites (e.g., Livejournal[5]), and social networking sites (e.g., Facebook[6], LinkedIn[7], MySpace[8]). At any time, social network users can create new accounts on various social networks and establish new social connections. This is consistent with the common behavior of social network users who are highly motivated to stay tuned with their contacts on each site and make use of the provided services and functionalities of each one. Note that a social network user is very likely to be involved in multiple social relationships with same real-world persons (co-referent) on different social network sites (e.g., a user can be connected to one of her friends on a personal social network site and on another professional site). Identifying co-referent users is important for enriching links with their corresponding labels; however, since we deal with different social network sites, it is often required to automatically design rules oriented accordingly. Thus, this calls for solutions to automatically identify co-referent users by taking into consideration the social network sites' characteristics, user profiles, and photos.

Obviously, the multiple and the different representations of the same physical persons across social network sites are one of the most intriguing problems. While studying such interactions across social networks, many challenges arise, urging the development of relevant methodologies and techniques to extract, generate, and infer correct information. In the following, we cite some of the challenges which make the task of identifying co-referent users difficult to achieve:

- **No data portability:** most of those sites prohibit the exchange of information and communication with other social networks such as sharing data, having contacts with a global profile, or interacting across different social network sites. Consequently, "People are getting sick of registering and re-declaring their friends on every site" [193]. This leads to unwanted information duplication, an increased spent time to refill the same information, and efforts to maintain each profile updated on each social network. In other words, social networks are functioning as "Data Isolated Islands" [151].

- **Absence of global unique identifier:** currently, social network sites enable users to describe their profiles using a set of attributes/properties, whereby properties defined as Inverse Functional Property (IFP) in the Web Ontology Language (OWL) [194] descriptions (such as email address, homepage, weblog, etc.) are only unique within each social network, none of them represents a globally unique identifier across different social network sites.

[4]http://www.flickr.com/
[5]http://www.livejournal.com/
[6]http://www.facebook.com/
[7]http://www.linkedin.com
[8]http://www.myspace.com/

- **Computational complexity:** due to the immense scale of social network sites and the large amount of social network contacts that a given user could be connected to, applying profile matching techniques over all pairs of records is computationally expensive requiring a total number of comparisons equal to the cross product of the size of the two sets of users. This makes the algorithm of at least a quadratic complexity because all profiles in the first set have to be compared to all profiles in the second set.

To handle the previously mentioned issues, new approaches that address the problem of entity resolution in the context of social networks are needed to tie the information across different sites and among various data types (mainly textual and multimedia). This presents a new challenge in order to alleviate the problem of a lacking methodology to generate rules as an important part to identify co-referent users according to specific networks' characteristics that need to be highlighted. The next section is specifically concerned with presenting a brief overview of our methodology able to generate co-referent rules.

5.3.1 Methodology Overview

This section briefly describes the methodology we propose to generate inter-social rules for the task of profile matching. As we mentioned previously, generating co-referent rules is crucial since there is no "globally unique identifier" across social network sites. This is why, it is necessary to provide appropriate rules to discover when two profiles refer to the same real-world person.

Unlike link type prediction where it is possible to use crowd-based techniques to generate intra-social basic rules, co-referent rules are only appropriate within each pair of social networks involved in the matching. Consequently, basic rules can be generated only when we already know the social networks to study.

We take this into account by proposing a methodology which is able to generate co-referent basic rules for two known social networks. Our basic rules generation methodology must be able to cope with two main problems:

1. **User Profile Domains:** even when social network sites share the same representation, user profile attribute domains are not always common. For instance, the domain values of *interest* attribute in Facebook do not necessarily meet the domain values of the same attribute in LinkedIn.

2. **Site/User Objectives:** depending on the site and on the user objectives, the same attribute can be filled in with different values. For instance, the email attribute in Facebook

is commonly filled with a personal email while the LinkedIn email is assigned to the professional email of the same user.

We distinguish two groups of basic rules:

- **Attribute-based basic rules:** identify co-referent users when the similarity score between each pair of attributes is above a specified threshold
- **Profile-based basic rules:** identify co-referent users when the global similarity score (e.g., the average of the similarity scores) between all the profiles' attributes is above a specified threshold.

Note that similarity metrics and decision making algorithms are needed to generate the set of basic rules. In what follows, we explain how we proceed to generate the inter-social basic rules.

5.3.2 Basic Rules Generation Methodology

We propose to generate the set of basic rules able to identify the co-referent relationship type. We describe the different steps including how to leverage IFP attribute(s) within our approach in order to generate the set of rules. In fact, while using an IFP (such as the email address) as a unique identifier(s) within a single social network is possible, it cannot be considered as a relevant global unique identifier across different social networks due to various types of constraints, i.e., different IFP attributes, missing IFP values, etc. Therefore, our methodology consists of generating basic rules by using available pairs of profiles having the same IFP values. Consequently, the set of generated basic rules reflects the characteristics of the studied social networks, and thus enabling the identification of remaining pairs of co-referent users having different IFP values, missing IFP values, or different IFP attributes.

The principal steps of our methodology, illustrated in Figure 5.2, can be summarized as follows:

1. First, we gather the pair of co-referent users, obtained using the specified IFP attribute(s).
2. Second, we generate the basic rules from the obtained set of co-referent users by analyzing their other attributes and thus creating the attribute-based and profile-based basic rules.

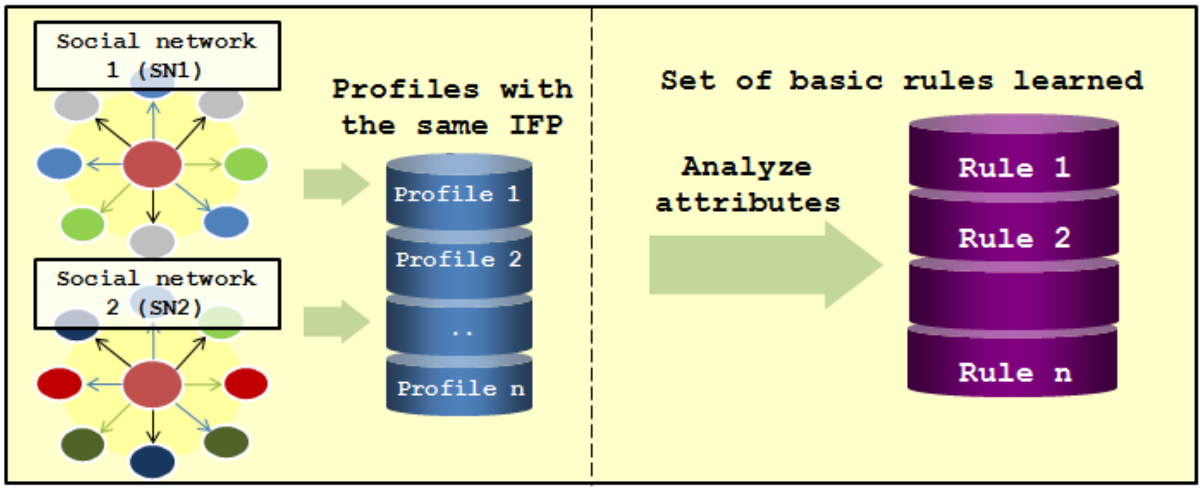


Figure 5.2: Our basic rules generation methodology composed of two parts: 1) selecting the profiles with the same IFP, and 2) generating the set of basic rules.

5.3.2.1 Choosing the Matching Attribute

Identifying co-referent users consists of retrieving users by selecting pairs of profiles having the same unique identifier value(s) (IFP). In our work, we consider *foaf:mbox_sha1sum* as default IFP since it is defined in the FOAF vocabulary adopted as a common representation of social profiles in our study. However, it is possible to choose another attribute as default IFP (by modifying this in the user preferences module c.f. Chapter 4). For instance, instead of using *foaf:mbox_sha1sum*, one may choose the *foaf:image* which represents the profile picture available commonly on social network profiles.

5.3.2.2 Generating Attribute-based Basic Rules

As mentioned previously, attribute-based rules are composed of several attributes and their corresponding weights that are used to find co-referent users. By computing the similarity between the attributes of all pairs of co-referent profiles (gathered in the previous step), we can obtain as many rules as the number of co-referent users found (based on the predefined IFP attribute(s)) where for each rule:

- **The body:** is composed of common[9] related attributes available in the profiles of a pair of co-referent users. A similarity score, obtained by computing the similarity score between

[9]Note that only common attributes are considered since determining correspondences between different attributes is out of the scope of our work.

Table 5.6: Example of a set of attribute-based basic rules.

BasicRule1:
(profileID1.name.$TextSimilarity_{metric}$(profileID2.name) >= 0.8) AND
(profileID1.lastname.$TextSimilarity_{metric}$(profileID2.lastname) >= 0.7) AND
(profileID1.nickname.$TextSimilarity_{metric}$(profileID2.nickname) >= 0.9)
→ (**co-referent**, 1)

BasicRule2:
(profileID1.name.$TextSimilarity_{metric}$(profileID2.name) >= 0.8) AND
(profileID1.lastname.$TextSimilarity_{metric}$(profileID2.lastname) >= 0.5) AND
(profileID1.image.$TextSimilarity_{metric}$(profileID2.image) >= 0.7)
→ (**co-referent**, 1)

BasicRule3:
(profileID1.name.$TextSimilarity_{metric}$(profileID2.name) >= 0.8) AND
(profileID1.image.$TextSimilarity_{metric}$(profileID2.image) >= 0.7) AND
(profileID1.interest.$TextSimilarity_{metric}$(profileID2.intereset) >= 0.9)
→(**co-referent**, 1)

BasicRule4:
(profileID1.name.$TextSimilarity_{metric}$(profileID2.name) >= 0.8) AND
(profileID1.lastname.$TextSimilarity_{metric}$(profileID2.lastname) >= 0.7) AND
(profileID1.nickname.$TextSimilarity_{metric}$(profileID2.nickname) >= 0.9)
→ (**co-referent**, 1)

each of these related attributes, is assigned to each common attribute. This similarity score is used as a similarity threshold. In order to identify co-referent users, each attribute in the rule's body must have a similarity score above that already computed similarity threshold.

- **The head:** always indicates the relationship *co-referent users* with a score equal to 1 ($\langle co-referent, 1\rangle$).

For instance, Table 5.6 shows some attribute-based basic rules examples. These basic rules, state that two users will be considered as co-referent if each of their attributes' similarity is above the specified similarity threshold. This process can be regarded as a supervised learning where FOAF attributes are given different weights, instead of being all equal.

To compute the similarity score between two common attributes, it is important to choose the appropriate metric for each attribute. In the following, we present several similarity metrics having different characteristics, and we discuss the most appropriate ones to use as default

similarity metrics.

Similarity Metrics

In order to obtain appropriate results, adapted similarity function(s) must be associated to each attribute (e.g., comparing `emails` must be computed in a different way than comparing `interests`). In addition, a profile can contain multimedia data in particular photos. Thus, visual similarity techniques must also be used to compare such data.

In the following, we detail several metrics used to measure the similarity between different profile attributes, then we present the default similarity metrics that we adopted in this work.

Textual and Numerical metrics

Various techniques can be used to measure the similarity score between two values and can be grouped into two main categories:

- **Syntactic-based similarity approaches**: provide exact or approximate lexical matching of two values. Using exact similarity techniques [195] can lead to poor similarity results since frequent variations of a word exist and typing errors are common. Thus, approximate string matching techniques [196] can be used to compute the distance between two values that have a limited number of different characters.
- **Semantic-based similarity approaches**: are used to measure how two values, lexically different, are semantically similar. In fact, lexical-based text similarity cannot identify semantic similarity. The semantic similarity between two texts has been measured using different techniques [197]. They can be:
  - Knowledge-based [198]: computing similarity between values with the usage of predefined (or external) knowledge resources (taxonomies, ontologies, etc.) such as WordNet [191], Wikipedia[10], etc. The similarity can be edge-based (computed following the distance separating values to be compared in the external knowledge) or node-based (computed following the amount of information that a concept contains).
  - Corpus-based [199]: computing the similarity between two concepts using large corpora only (and without external knowledge resources). The similarity can be based on vector-space model [200], statistical such as Pointwise Mutual Information retrieval [201], or Latent Semantic Analysis [202].

[10] http://www.wikipedia.org/

Visual metrics

As for the visual similarity between images, several features can be used to compute the similarity. Visual features include color, edge, texture, shape, etc. Each feature has its own representation (vector, histogram, etc.), and has its corresponding matching technique(s) (e.g., Euclidean distance, hamming metric). Broadly speaking, image similarity techniques can be categorized into two groups [203]:

- **Global measures approaches:** return a single output score that represents how much similar two images are. They include different measures such as probabilistic approaches [204], χ^2-based distance measures [205], and cross-bin distance measures [206].
- **Local measures approaches:** return a similarity image that represents the local similarity between two images. They include approaches such as mean square error [207], mutual information [208], and cross-correlation coefficient [209].

In the following, we present the similarity metrics that we adopted as default.

Default Similarity Metrics

Assigning default similarity functions to FOAF attributes must be done carefully with respect to the attributes' characteristics and the domain values. Figure 5.3 summarizes our proposed default similarity measure assignments to FOAF attributes. In the following, we detail these metrics.

Textual and Numerical metrics

The different similarity measures that we use are:

1. **Senseless One-term attributes:** as stated in [176] [210], Jaro metric [132] is considered as one of the optimal measures to be primarily intended for short string comparison. It is based on the number and the order of the common characters between two strings. The definition of common characters is that the agreeing characters must be within half of the length of the shorter string. The Jaro distance similarity between two strings s and t can be computed as follows:

$$sim_{Jaro}(s,t) = \frac{1}{3}\left(\frac{|s'|}{|s|} + \frac{|t'|}{|t|} + \frac{|s'| - 0.5 \times T_{s',t'}}{s'}\right), \tag{5.1}$$

where:

- $|s|$ and $|t|$ are the length of each string,

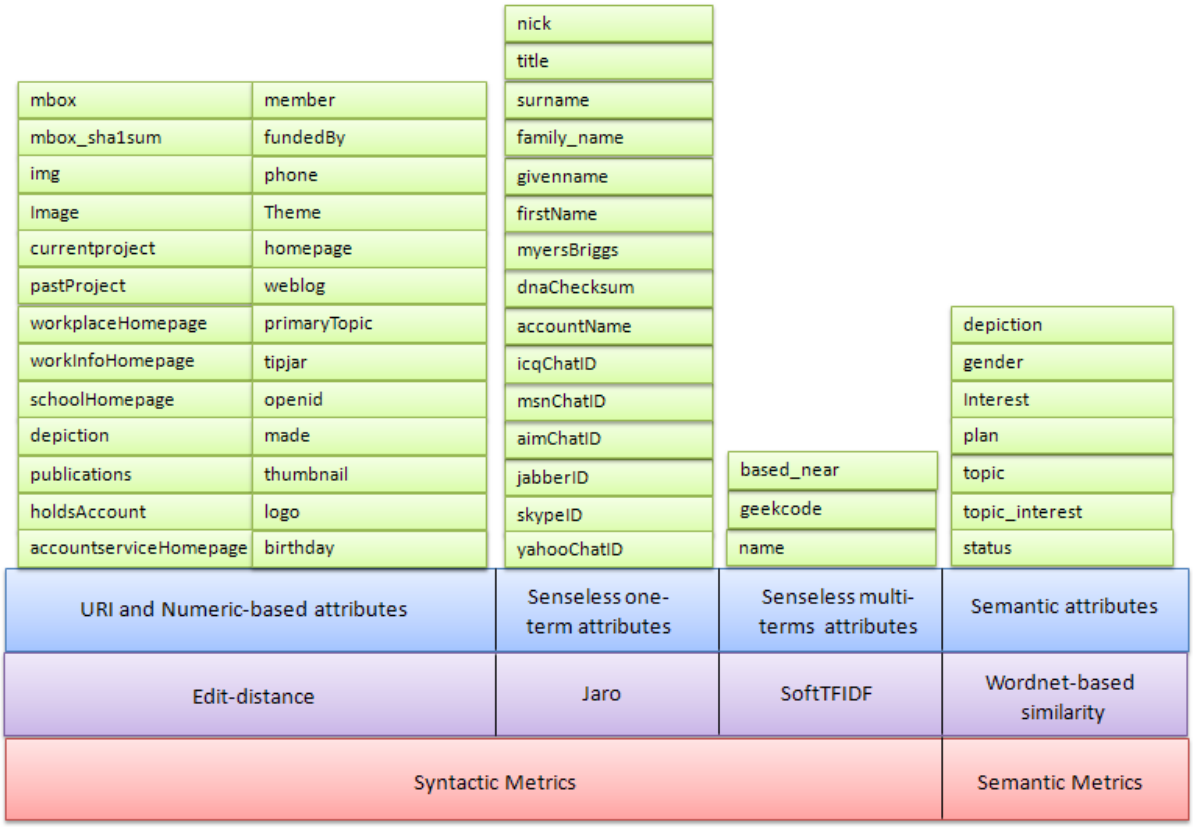


Figure 5.3: Default metrics to compute similarity of each FOAF attribute

- $|s'|$ and $|t'|$ are the number of common characters,
- T is the number of transposed characters.

2. **Senseless Multi-terms attributes:** the SoftTFIDF metric [176] is one of the best techniques [211] that combine the token-based (or words) and string-based methods to compute similarity between sentences. It is based on the cosine similarity that doesn't automatically discard words which are not strictly identical. This metric has two main advantages: 1) the token order is not important, and 2) common uninformative words don't greatly affect similarity [211] [212]. The SoftTFIDF similarity measure between s and t can be computed as follows:

$$Sim_{SoftTFIDF}(s,t) = \sum_{w \in close(\phi,s,t)} V(w,s) \times V(w,t) \times D(w,t), \tag{5.2}$$

where $close(\phi, s, t)$ is the set of words $w \in s$ such that there are some $v \in t$ and the

distance $dist'(w, v) \geq \phi$, and for $w \in close(\phi, s, t)$, $D(w, t) = max_{v \in t} dist(w, v)$.

3. **Semantic-based attributes:** the Explicit Semantic Analysis (ESA) is a technique that uses Wikipedia to compute semantic relatedness and is considered as one of the best existing methods [177]. Each concept is represented by a weighted vector that contains a text describing the concept with a weight computed using the TF-IDF measure. A semantic interpreter is formed by all the concepts and their weighted terms. It tries to match each word to the most relevant concepts based on a defined threshold. For a more efficient search, a constructed inverse interpreter index maps each word to all the concepts that are part of them. A weighted vector that represents the relevance of the concepts to a vector with a weight is calculated for each text snippet. At the end, the cosine measure is applied to the two vectors so to compute the relatedness between the two text snippets.

4. **URI and Numeric-based attributes:** the Edit Distance (ED) metric [131] is the most suited technique to compute similarity for URI and numeric-based attributes. It measures the distance between two strings/numbers, s and t, by calculating the cost of the minimum number of editing operations (insertions, deletions, and substitutions), commonly called edit script, that convert s to t. The edit distance similarity between two values s and t can be computed as follows:

$$sim_{EditDistance}(s, t) = 1 - \frac{d}{max(l_s, l_t)}, \tag{5.3}$$

where:

- s and t: the two values to compare,
- d: the distance (cost) between s and t,
- l_s and l_t: the length of s and t respectively,
- $max(l_s,l_t)$: the maximum length between s and t.

To illustrate this, we applied the default metrics given above to compute the similarity between the attributes of two sample profiles provided in Figure 5.4 belonging to the same person.

Table 5.7 shows the obtained similarity scores. One can see that default metrics provide the best similarity scores.

Visual metrics

To measure the visual similarity between two images, we propose to use a global measure technique. In fact, we are interested in approaches that return a similarity value between two

```
<foaf:Person>
<foaf:name>Alexandre William</foaf:name>
<foaf:firstname>Alexandre</foaf:firstname>
<foaf:family_name>William</foaf:family_name>
<foaf:img>www.xyz.com/alex/photos/alex.jpg</foaf:img>
<foaf:interest>Paris, Software, Internet</foaf:interest>
</foaf:Person>
```

Profile 1

Profile 2

```
<foaf:Person>
<foaf:name>William, Alexandre</foaf:name>
<foaf:firstname>Aleksandre</foaf:firstname>
<foaf:family_name>Willlaim</foaf:family_name>
<foaf:img>www.abc.com/photos/aleks.jpg</foaf:img>
<foaf:interest>France, IBM, Web</foaf:interest>
</foaf:Person>
```

Figure 5.4: Two sample FOAF user profiles

Table 5.7: Similarity scores of Profile 1 and Profile 2 using default similarity metrics

Attributes/Similarity Metrics	Jaro	ED	SoftTFIDF	ESA
$< foaf : name >$	0.72	0.12	**0.99**	0
$< foaf : firstname >$	**0.85**	0.6	0.85	0
$< foaf : img >$	0.77	**0.8**	0.66	0
$< foaf : interest >$	0.52	0.22	0	**0.75**

images instead of returning a similarity image which represents the local similarity between the two input images. We adopted a visual descriptor called "Color and Edge Directivity Descriptor" (CEDD) [213]. Our choice is based on two reasons: 1) CEDD has a low computational power, and 2) it is comparable with the needs of the most MPEG-7 descriptors. This compact descriptor uses both of the color histogram and the texture features histogram. To measure the distance between two images, CEDD uses the Tanimoto coefficient [214]:

$$T_{i,j} = t(x_i, x_j) = \frac{x_i^T x_j}{x_i^T x_i + x_j^T x_j + x_i^T x_j}, \tag{5.4}$$

where:

- x_i and x_j are the two images' vectors,
- x^T is the transpose vector of x.

5.3.2.3 Generating Profile-based Basic Rule

Attribute-based basic rules may not be able to identify all co-referent users. In fact, when comparing the attributes of two profiles, the similarity values between all common attributes

Table 5.8: Example of a set of profile-based basic rules.

BasicRule5: U1.ProfileSim(u2) > th_ProfileMatching → (**co-referent**, 1)

must be above the specified threshold of the same corresponding attributes at least in one attribute-based basic rule. These rules can be restrictive since it is enough that there exists a pair of co-referent users that agree on all the common attributes except one attribute.

As some of the co-referent users may not be identified by using the set of basic rules, we propose to use the profile-based basic rules to identify co-referent users that have not been identified yet. Therefore, we propose to consider the global profile similarity when comparing two profiles. Consequently, whenever two profiles have an aggregated similarity above a predefined threshold, they are considered as co-referent. A profile-based basic rule is shown in Table 5.8

In the following, we detail the methodology to create the profile-based basic rule that consists of several steps: 1) computing the attributes' weights of users' profiles, 2) showing how to define the profile matching threshold which is used to decide whether two profiles refer to the same physical user or not, and 3) computing similarity between profiles.

Assigning Weights to FOAF Attributes

To identify correctly remaining co-referent users, attributes must be assigned weights since they do not contribute equally in the task of identification. For instance, the first name and the last name attributes can be more important than the gender attribute between two studied social networks. The choice of the most appropriate weight to each FOAF attribute requires comparing attributes similarity, considering different co-referent users, all of which are gathered (as previously explained earlier) based on their IFP values. Assigning a weight to each FOAF attribute is very important since attributes' importance varies depending on the two studied social networks.

In our work, the weight can be assigned manually or computed automatically. Manual assignment allows users to include their preferences and inputs in the matching process (e.g., *mbox* attribute may be the most important for a user) while automatic assignment is provided in order to allow considering related social network characteristics (e.g., *homepage* attribute is more important on LinkedIn than on Facebook). Of course, it is possible to use both types of assignment (one can start with automatic assignment and tune it manually after receiving the weights).

To assign weights automatically, our methodology processes each pair of the previously

gathered co-referent users. The common FOAF attributes of each pair are compared together and a similarity score is computed. Each time a similarity score is obtained, it is associated to its corresponding FOAF attribute. Once comparing all pairs of co-referent profiles is done, each FOAF attribute possesses a set of associated similarity scores. Using these associated similarity scores, we compute each attribute's final weight as the average of the associated similarity scores. Note that it is possible to use, instead of the average, other functions or techniques according to different users' requirements.

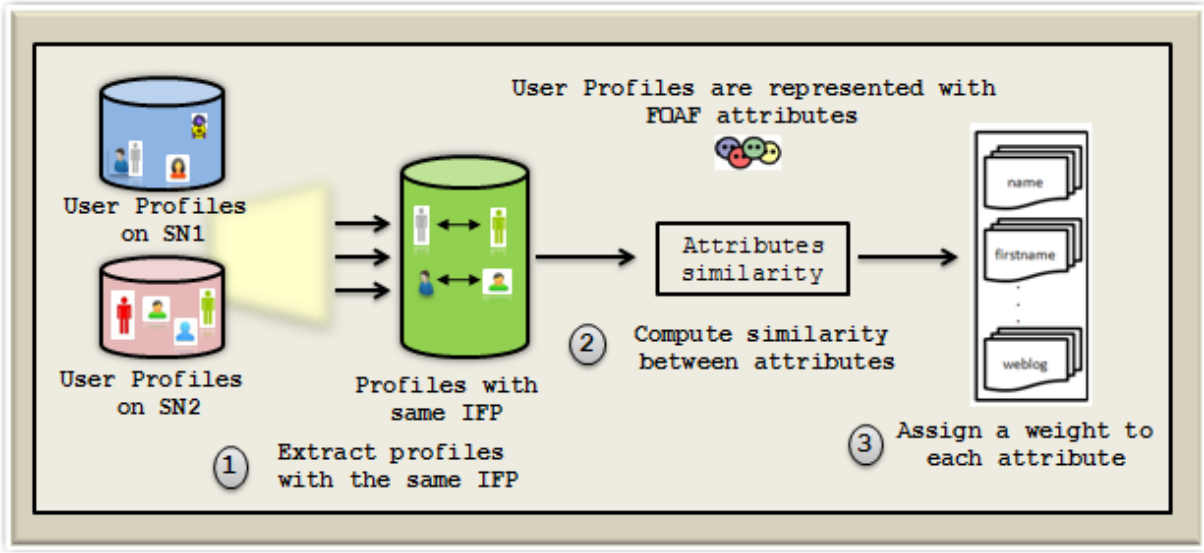


Figure 5.5: Assigning weights to FOAF profiles.

Defining the Profile Matching Threshold

The profile matching threshold defines the minimal similarity value required for matching two profiles. We compute this threshold using the weights assigned to each attribute as shown previously. The assumption here is that these weights are the result of the aggregation of values coming from different profiles that refer to same real-world users. Based on this, weights form reliable measures and can be considered as reference values for computing a profile matching threshold. More details about aggregation using decision making approaches is provided in Appendix A. Here, an aggregation function is needed to compute this threshold score which is represented as follows:

$$th = f_{decision}\left(w_0\left(a_0\right), w_1\left(a_1\right), .., w_n\left(a_n\right)\right)$$

where:

- th: the profile matching threshold to compute,

- $f_{decision}$: the decision making algorithm used to return the aggregated value,
- a: an attribute used to describe a user profile,
- n: the number of available attributes,
- w: the weight assigned to an attribute.

Computing the Global Profile Similarity Score

To identify whether two profiles refer to the same real-world users or not, the similarity scores between their attributes' values are computed as follows:

1. First, the values of common attributes in both profiles are extracted and their similarity scores are computed,
2. Second, the obtained similarity scores are tuned, using attributes' weight, in order to have more realistic scores that take into consideration the importance assigned to each attribute,
3. Third, a global similarity score is computed using an aggregation function.

Taking into account the weights assigned to attributes, whenever two attributes are compared, the obtained similarity score between two attributes' value will tend to increase or decrease depending on the importance/weight associated to the given attribute. The new similarity scores between two attributes take into account 1) the similarity score between the attribute's values, and 2) the weight assigned to the attribute. This is computed as follows:

$$sim'\left(P1.a_i, P2.a_i\right) = \frac{2 \times sim\left(P1.a_i, P2.a_i\right) \times w\left(a_i\right)}{1 + \left(sim\left(P1.a_i, P2.a_i\right) \times w\left(a_i\right)\right)} \tag{5.5}$$

where:

- a_i: an attribute used to describe a profile,
- $P1.a_i$ and $P2.a_i$: two values of an attribute a_i in Profile P1 and Profile P2,
- $w\left(a_i\right) \in [0, 1]$: the computed/assigned weight of an attribute a_i,
- $sim\left(P1.a_i, P2.a_i\right) \in [0, 1]$: the initial similarity score computed between the values of an attribute in P1 and P2,
- $sim'\left(P1.a_i, P2.a_i\right) \in [0, 1]$: the final similarity score computed between the values of an attribute in P1 and P2.

Once all final similarity scores of all attributes are computed, these scores are sent to an aggregation function. The obtained score represents the global profile similarity score. Whenever this score is above the profile matching threshold, then the two users are considered as co-referent. For instance, a profile-based basic rule states that two users are considered as co-referent if the similarity between their profiles is above the "profile matching threshold" previously explained. To choose an adapted aggregation function, we surveyed different existing methods [215]. A summary of these different approaches is detailed in the following.

Decision Making Algorithms

An aggregation function aims to reduce uncertainty by filtering and/or aggregating a set of values in order to select or compute one relevant value to facilitate decision-making. Since making decision based on single value is sometimes prone to errors, therefore a better accuracy is achieved whenever it is possible to combine data from different sources to improve the decision making process. Here, an aggregation function, as shown in Figure 5.6, outputs a single value as a result of a set of inputs.

Among the most known decision and probabilistic analysis techniques, one can mention the probabilistic methods (e.g., Bayesian networks [216]), the evidence theories (e.g., Dempster-Shafer [174] [175]), the fuzzy set theories (e.g., fuzzy decision trees [217]), and other classical functions (e.g., average, minimum, maximum, etc.) [218] [219]. The classical functions have been studied and used extensively in many fields (e.g. database, multimedia, security, etc.). In the following, we will briefly present the Bayesian network, the Dempster-Shafer theory and fuzzy decision trees.

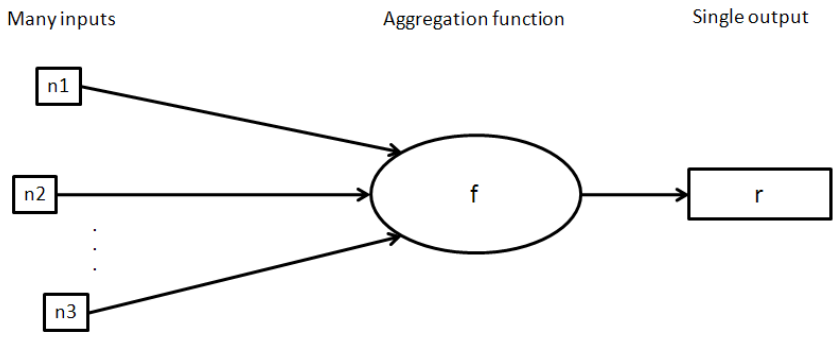


Figure 5.6: An aggregation function.

Default Decision Making Algorithms

We choose to use Dempster-Shafer as the default algorithm based on the following reasons:

1. The Minimum and Maximum functions are not suitable to our objective. On social network, it is very common to have two profiles with different attributes' values while referring to the same real-world person. Using the Minimum functions will prevent the system from identifying the corresponding users as co-referent. In fact, this is too restrictive since it is likely that the obtained minimum similarity has a score below the defined profile matching threshold value. At the same time, two social network profiles that refer to different real-world persons can have one or more attribute(s) with high similarity. Using the Maximum functions will directly consider the corresponding users as co-referent, which is not correct. In fact, this can lead to unreliable results since it is very probable that the maximum similarity score has a value above the profile matching threshold.

2. The fuzzy decision tree methods are only used to compute the ambiguity/precision of an attribute. Ambiguity as computed using the fuzzy decision trees can be subject to significant change of values for a small change in the input data. This is not suitable for social network sites where data is dynamic and changes frequently.

3. One of the main complications in BNs is when a new evidence is added, the probabilities at each node are recomputed to propagate the evidence through the nodes. Another drawback for BNs is that they cannot distinguish between the lack of evidence for a proposition and the evidence against the proposition.

4. Meanwhile DS theory can make the difference between the lack of evidence for a proposition and the evidence against the proposition [218]. In fact, this advantage of DS is the result of the non existence of a causal relationship between a hypothesis and its negation, so the lack of belief does not imply disbelief. The DS theory is able to represent both imprecision and uncertainty, flexibility, and its ability to consider more than one class for decision making. In the following, we explain how to compute the profile matching threshold and the similarity score between two profiles.

5.3.3 Discussion

Inter-social networks rules are required in several scenarios (data integration, data enrichment, information retrieval, etc.). In this work, we use inter-social rules to achieve entity resolution on social networks. This means that the rules are used to identify co-referent users among the contacts of the same user across different social networks.

In this work, we address the problem of matching user profiles in its globality by providing a methodology to automatically generate basic rules. With a suitable matching framework able

to consider all the profile's attributes, the generated basic rules are grouped into two categories: 1) attribute-based basic rules, and 2) profile-based basic rules. Rules are iteratively applied, first the attribute-based basic rules and then the profile-based basic rules. Different similarity and decision making algorithms are used in this methodology.

Our contributions consist of:

1. Describing a methodology to generate rules in order to identify co-referent users across different social networks. Obtained rules reflect the characteristics of the different social networks analyzed. These two types of basic rules are: 1) attribute-based basic rules and 2) profile-based basic rules
2. Assigning attributes with corresponding weights within the profile-based basic rules.
3. Generated rules take into account all the profiles' attributes (textual and photos data).

Nowadays, social network users differ in terms of profile information and in the way they maintain and update their profiles on different social networks. So far, the rules that we propose to generate are part of a wider set of rules that are user-oriented. Therefore, there is a need to make these inter-social rules less restrictive by taking into account a large set of cases. Naturally, this consists of extending the sets of rules and help toward achieving their completeness. We discuss this in the next section.

5.4 Derived Rules

The previous sections presented the sets of basic rules for both social and co-referent relationship discovery. We showed how to generate each of these two sets of basic rules. Although these basic rules aim to discover different relationship types, a set of relationship may remain unlabeled. On social network sites, the sets of rules that take into account users' profiles and their related data are certainly the most relevant to a user. To that end, we present in this section an additional type of rules that we call *derived rules*. Derived rules are generated using a mining technique which successfully generates rules relevant to the context of a main user.

Derived rules are viewed as hidden knowledge that can be formed by the correlation between attributes (user profiles and photos' related attributes) and relationship types (colleagues, relatives, friends, etc.). They have the same form of rules (c.f. Chapter 4), and are made of a body and a head. However, they combine rules' body conditions differently. Indeed, derived rules may contain a different number of conditions and are assigned a score (a confidence and support

score) that can be used to measure the likelihood that a rule is true or not. For instance, rules with low confidence and low support scores can be filtered since they may not be reliable enough to be used.

Our approach leverages the characteristics and features of social networks, namely by using, on one hand, the available information within users' profiles and, on the other hand, the embedded information extracted from shared photos in which users' contacts are depicted so to measure and qualify the types of relationships between users. Derived rules are of considerable importance to remedy the problem of rule incompleteness. They can discover all interesting associations of data attributes that are able to generate further rules. To generate these rules, classification association rules, a special subset of association rule mining [220], are used in this work. Classification using association rules are composed of two fundamental parts: association rule mining, and classification association rule. We give a brief introduction about Association Rule Mining and Classification Association Rules in Appendix B. Both intra-social basic rules and inter-social basic rules can be extended using a well-known mining technique, namely the Apriori algorithm [12].

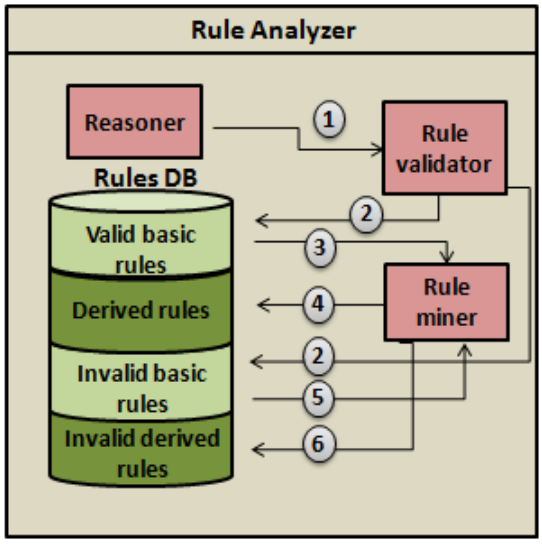


Figure 5.7: Our derived rules generation methodology.

To generate the set of derived rules, the rule analyzer, shown in Figure 5.7, is used. The rule analyzer receives a set of basic rules related to the corresponding relationship discovery task (social or co-referent discovery). As shown in Figure 5.7, the rule analyzer starts (step 1) by testing the validity of the set of basic rules over the stored photos (social relationship discovery) or user profiles (co-referent relationship discovery) in order to come up with *valid basic rules* and *invalid basic rules* (step 2). In the context of social relationship discovery, a valid rule

is a basic rules that matches with at least one photo, otherwise it is considered as an invalid rule and is stored within the set of *invalid basic rules*. Similarly, in the context of co-referent relationship discovery, a valid rule is also a basic rule that is able to detect at least one pair of co-referent users (within the user's contacts). In fact, the set of valid rules contains all rules having their body's conditions satisfied when applied on a given user profile. Consequently, while extending the set of rules (step 3), only the *valid basic rules* are mined (step 4). The obtained rules are called the *derived rules*. For a different reason, the set of *invalid basic rules* is also mined (step 5). The obtained rules, called *invalid derived rules*, don't reflect the context of the user (step 6). However, this means that the rule analyzer is able to detect the existence of undefined relationship types and thus allows the user to label them. Accordingly, the rule analyzer suggests the possibility of having a set of photos that shares common patterns.

In the next section, we present our relationship discovery algorithm that underlines the use of the basic and derived rules to identify social and co-referent relationships between social network users.

5.5 Algorithm

In the following, we present the algorithm of our relationship discovery, which seeks to generate the appropriate rules to reveal all interesting, yet not generated, rules based on the context of each user using the set of valid basic rules. In addition, we detail all the steps in order to explain how the basic, and extended rules are used all together in this algorithm. At the end of this section, we present an optimization technique to enhance the co-referent relationship discovery.

We provide the pseudo-code of our relationship discovery as shown in Algorithm 3. This algorithm uses intra-social basic rules that target photos for the discovery of social relationships, and inter-social rules that target user profiles for the discovery of co-referent users. The algorithm takes into account the user's choice that consists in choosing whether to discover social or co-referent relationships. In Algorithm 3, social and co-referent relationship discovery are listed below:

- Social relationship discovery: rules are applied on photos
- Co-referent relationship discovery: rules are applied on profiles

Here, we explain the different steps described in Algorithm 3:

Step 1 (line 5): as shown in Figure 5.7, the algorithm tests, via its "Rule validator" component,

Algorithm 3: **Relationship Discovery Steps**

Input:
$sn_n.u_0$, // The profile of the main user on a social network
1 $sn_n.u_0.PhotoProfiles[]$, // The set of photos and profiles of u_0 on a social network
2 $sn_n.u_0.basicRules[]$, // The set of basic rules
3 RelToIdentify // The relationship type to identify: 1) social or 2) co-referent
Output:
$sn_n.u_0$ // The labeled social network(s) of the user u_0
4 **begin**
// Step 1: Test the set of basic rules
5 **if** *RelToIdentify == "social"* **then**
// Discover social relationships within a single social network.
// AssignRelationship(RulesToTest, UserProfileSN1, PhotoProfiles[])
// is a function presented in Algorithm 4.
// It assigns a social relationship.
6 $sn_1.u_0 \longleftarrow$ AssignRelationship($basicRules[]$, $sn_1.u_0$, $PhotoProfiles[]$);
7 **else**
// Discover co-referent relationships across different social networks.
// AssignRelationship(RulesToTest,UserProfileSN1, UserProfileSN2,
// PhotoProfiles[]). It assigns a co-referent relationship type.
8 ($sn_1.u_0$,$sn_2.u_0$) $\longleftarrow$ AssignRelationship($basicRules[]$, $sn_1.u_0$, $sn_2.u_0$, PhotoProfiles[]);
9 **end**
// Step 2: Mine the valid rules
10 **if** *RemainingPhotosProfiles not empty* **then**
// Apply mining technique on the set of valid basic rules to generate
// a set of derived rules and test this new set of rules
11 DerivedValidRules[] $\longleftarrow$ Apriori(ValidRules[]) ;
12 **if** *RelToIdentify == "social"* **then**
13 AssignRelationship(DerivedValidRules[], $sn_1.u_0$, RemainingPhotosProfiles[]);
14 **else**
15 AssignRelationship(DerivedValidRules[], $sn_1.u_0$, $sn_2.u_0$, RemainingPhotosProfiles[]);
16 **end**
17 **end**
// Step 3: Mine the invalid rules
18 **if** *RemainingPhotosProfiles not empty* **then**
// Apply mining technique on the set of invalid basic rules to generate
// a set of derived invalid rules and test this new set of rules
19 DerivedInvalidRules[] $\longleftarrow$ Apriori(InvalidRules[]) ;
20 **if** *RelToIdentify == "social"* **then**
21 AssignRelationship(DerivedInvalidRules[], $sn_1.u_0$, RemainingPhotosProfiles[]);
22 **else**
23 AssignRelationship(DerivedInvalidRules[], $sn_1.u_0$, $sn_2.u_0$, RemainingPhotosProfiles[]);
24 **end**
25 **end**
26 Return $sn_n.u_0$;
27 **end**

the validity of the set of basic rules over the stored user profiles and photos. When a rule matches with at least one photo or pair of profiles, the indicated relationship in the rule is assigned to the corresponding contacts. The rule is then considered as a valid basic rule.

Algorithm 4: **Assign Social and Co-referent Relationships**

```
   Input:   // The inputs from Algorithm 3
1    sn_n.u_0,  // The profile of the main user
2    RulesToTest[],  // The set of the actual rules to test from
3    PhotosProfiles[]  // The set of actual photos and profiles to use with the rules
   Output:
   R(U, V)  // The relationship type between corresponding social network users.
   // For social relationship discovery R is between the main user and one of her contacts.
   // For co-referent relationship discovery R is between two contacts of a main user
4  begin
       // The validity of each rule is tested over the set of photos (social relationship
       // discovery) and user profiles (co-referent relationship discovery)
5      foreach bRule_i in RulesToTest do
6          foreach PhotosProfiles_j in PhotosProfiles do
               // Test the validity of the rule bRule_i
7              if CheckRuleValidity(bRule_i) then
                   // Stores the set of photos/profiles that match at least one rule
8                  UsedPhotosProfiles[] ⟵ PhotosProfiles_j ;
                   // Stores the rule in the Rules DB as valid
9                  ValidRules[] ⟵ bRule_i ;
                   // Retrieves the names of the persons in the photo
10                 PersonsInPhoto[] ⟵ PhotosProfiles_j.GetPersonNames() ;
11             else
                   // Stores the rule in the Rules DB as invalid
12                 InvalidRules[] ⟵ bRule_i ;
                   // Stores the set of photos/profiles that didn't match any rule
13                 RemainingPhotosProfiles[] ⟵ PhotosProfiles_j ;
14             end
15         end
16     end
17     Return R(U,V);
18 end
```

Step 2 (line 10): After applying the set of basic rules, it is possible that some relationships remain unlabeled. Therefore, the set of "Valid rules" must then be mined with respect to the main user's context. The valid basic rules are sent to the rule miner which processes them using the Apriori algorithm. In return, the miner outputs the set of derived valid rules. The derived

valid rules are then applied to identify yet undiscovered relationship between users.

Step 3 (line 18): The set of *invalid basic rules* is then mined with respect to the main user's context. These rules are sent to the rule miner that returns the set of *invalid derived rules*. These rules reflect a general pattern describing the set of photos/profiles that didn't match any basic rule.

Note that relationships assigned to contacts must have a score above a defined threshold, otherwise the relationship type is removed since it is not considered enough trustworthy.

Optimization

To avoid a prohibitively expensive comparison of all pair of profiles, we adopt the *blocking* technique [221] applied in record linkage for the task of identifying co-referent users.

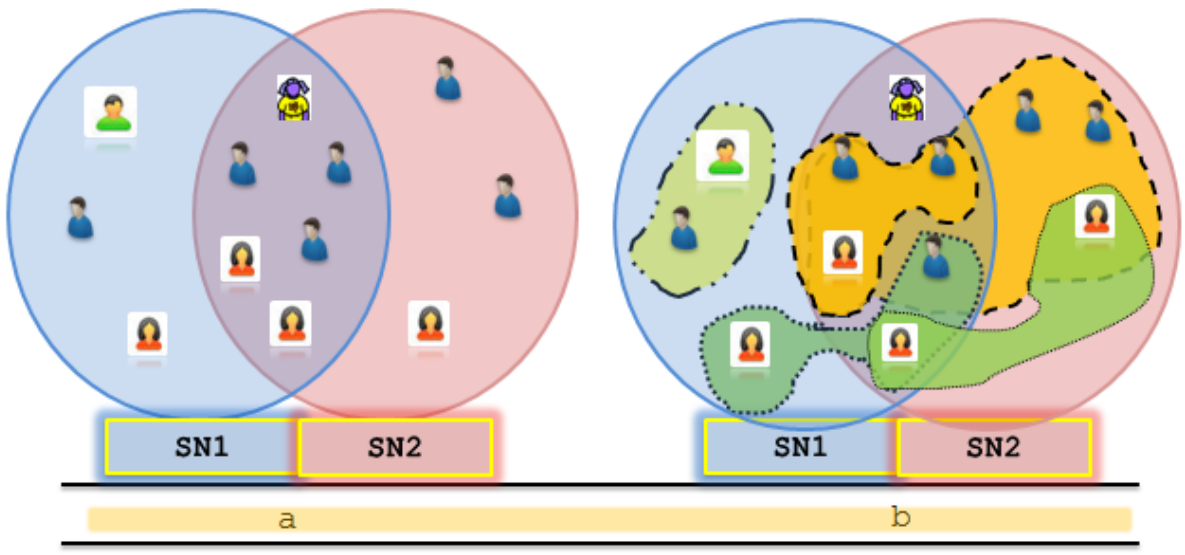


Figure 5.8: Social Network of Bob within the SN1 and SN2.

As illustrated in Figure 5.8, only blocks of users assigned with the same relationship type are compared together in the first phase (e.g., colleagues of SN1 with colleagues of SN2, relatives of SN1 with relatives of SN2, etc.). However, it is possible that certain contacts are assigned to multiple relationship types across different social network sites. Therefore, users with multiple and different relationship types are also compared together in the second phase (e.g., colleagues of SN1 with friends of SN2, colleagues of SN1 with relatives of SN2, etc.). Co-referent users found in the first phase are excluded from the set of users to compare in the second phase. It is important to note that in the worst case, all blocks are compared together which goes back to

the case where the "blocking" technique is not used. We therefore suggest to identify co-referent users only after the discovery of social relationship types.

5.6 Summary

In this chapter, we have presented our relationship discovery approach. We described the different types of rules that we propose. We first detailed the basic ones which are used to identify social relationships or co-referent relationships. Then, we showed how to mine the set of basic rules to obtain the derived rules.

As discussed above, derived rules are useful to identify relationships that remain unlabeled after using the set of basic rules. In fact, when processing real-world data, the issue of information completeness is one of the major challenges. There are, of course, social network profiles and photos with missing information and social networks' users with different needs. Only using the set of basic rules may prevent the system from discovering the social and co-referent relationship types. Therefore, in addition to the basic rules, we must consider rules that can be derived based on the context of each social network user as it is described in this chapter. This would improve the relationship discovery results within the same or across different social network sites. Furthermore, since discovering co-referent relationship type can be a computationally demanding and time consuming task, we therefore propose the optimization of this task by using the blocking technique in an attempt to reduce its complexity.

The contribution of this chapter can be categorized as follows:

- We proposed a methodology to generate the set of social basic rules. Our methodology is based on a crowdsourcing scenario, using a constructed photo dataset. We proposed a methodology to automatically construct this photo dataset. We also identified the requirements to follow in order to build such a dataset. In addition, we listed the set of generated social basic rules we obtained by applying our proposed rules generation methodology.

- We proposed another methodology to generate the set of co-referent relationships. Our approach considers all the attributes of users' profiles to identify co-referent users across different social network sites. The basic rules that we propose are divided into two categories:

  - attribute-based basic rules: are based on measuring the similarity score between each pair of common attributes between two profiles.

  - profile-based basic rule: is based on computing the global similarity score between two profiles. The global similarity score of two co-referent users must be above a computed threshold (the profile matching threshold).

- We provided a method to mine the derived rules. This technique takes into consideration the profile of the users and applies the Apriori algorithm. In addition, we suggest an optimization technique to reduce the complexity and enhance the effectiveness of our relationship discovery approach.

Chapter 6

Prototype and Experimentations

ABSTRACT

RelTypeFinder is a prototype designed using C# in order to validate our relationship discovery approach. This prototype is composed of several modules, including profile retriever, profile generator, photo editor, preference manager, rule miner, rule generator, and relationship finder. In this chapter, we detail these modules and provide detailed experimental results, using both real and syntactic datasets, to demonstrate the efficiency of our methodologies to reliably generate rules and identify relationship types.

6.1 Prototype

This chapter presents our prototype called *RelTypeFinder*, which we developed to validate the feasibility of the different proposals made in this work. We designed and implemented *RelTypeFinder* using C# language.

RelTypeFinder allows to discover social and co-referent relationship types between users of one or two different social networks through the use of rules applied on users' profiles. Rules are generated using *RelTypeFinder* and extracted from user profiles and photos datasets.

To demonstrate the relevancy of the generated rules and test their efficiency to identify appropriately social and co-referent relationships, we conducted different kinds of tests on real and synthetic datasets starting from validating the correctness and efficiency of social basic rules, to proving the relevance of co-referent basic rules. Within these tests, we varied different parameters such as thresholds, number of profiles, number of photos, etc. We also compared the execution time and provided detailed result analysis.

In the following, we detail the different modules of *RelTypeFinder*, illustrate different screenshots, and highlight the prototype's functionalities before describing the set of conducted experiments.

6.1.1 Profile Retriever

The profile retriever module is used to extract the profile of a given social network user with its related data. We recall that a user profile, represented using FOAF vocabulary, is described with a set of attributes, including name, image, contacts, etc. Photos, which are commonly present within user profiles, are retrieved with their metadata, captions, descriptions, and salient objects as explained previously in Chapter 4.

Note that we collect data using the profile retriever through the use of Application Programming Interface (API) of some popular social network sites. By using API, the profile retriever module is able to download part or whole of the real-world data. The main user starts by providing her URI to the profile retriever which handles the task of verifying and authenticating users before collecting data. Extracted profile information are stored as RDF files, locally at the user's computer, along with their corresponding photos.

6.1.2 Profile Generator

The profile generator module, shown in Figure 6.1, allows to create synthetic datasets composed of user profiles. As explained later in this chapter, we use synthetic datasets to evaluate the co-referent relationship discovery approach. Using this module, one can generate profiles by varying different settings such as the number of the profiles to generate, the percentage of co-referent profiles, the percentage of common attributes between generated profiles, and the number of occurrences of each attribute's value. Indeed, to generate these profiles, the module needs to assign random values to each attribute based on the previously mentioned settings. In fact, generating large and different sets of profiles (for testing purposes) requires the availability of a large number of values associated to each attribute. To this end, the profile generator module provides the user with the possibility to generate a large number of values from a small seed. Note that each seed is associated to a different FOAF attribute. In fact, this is another functionality called *random word generator* included in the profile generator module. The user has to provide a small number of values, define the number of similar values to generate, and choose the minimum similarity score between the seed and its generated values. Note that each attribute is assigned with a default similarity measure as shown in Chapter 4. For instance, if a user wants to generate three words having a similarity of 0.9 with the seed word *happyman*, the word generator assigned with the edit distance metric may return the following words: *hapyman*, *happymn*, and *happiman*. Note that the profile generator module notifies the user whether the available (or generated) attributes' value are enough (or further values must be generated) to create the profiles as defined in the above-mentioned user's settings.

6.1.3 Photo Editor

The photo editor module, shown in Figure 6.2, provides the main user with the possibility to edit the photos' embedded metadata, tags, descriptions, etc. The main user starts by providing the path of the locally stored photos previously retrieved using the profile editor module. In fact, this module is quite useful since it allows to:

1. Check and complete missing attributes (the photo album title and photo description).

2. Edit the value of embedded Exif information (time, date, and GPS location).

3. Tag an additional person not previously depicted in the photo.

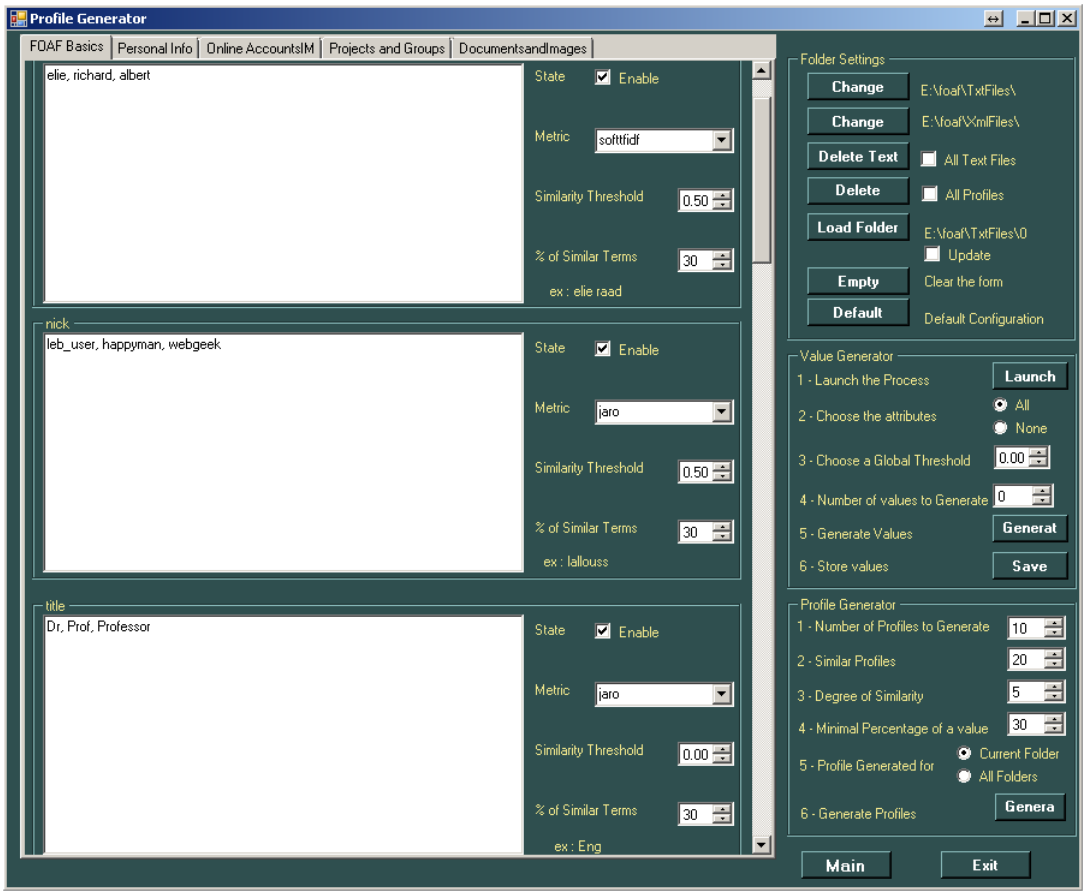


Figure 6.1: Screenshot of the profile generator module.

6.1.4 Preference Manager

This module allows a main user to define her own settings in terms of:

1. General preferences (user age category, relationship types to discover, age ranges categories)

2. Relationship-based preferences (data, time, location, and keywords related to a relationship)

3. System preferences (default IFP attribute(s), weights of profile's attributes, similarity metrics, and aggregation functions).

Figure 6.3 illustrates the preference manager module (the general and relationship-based preferences interface).

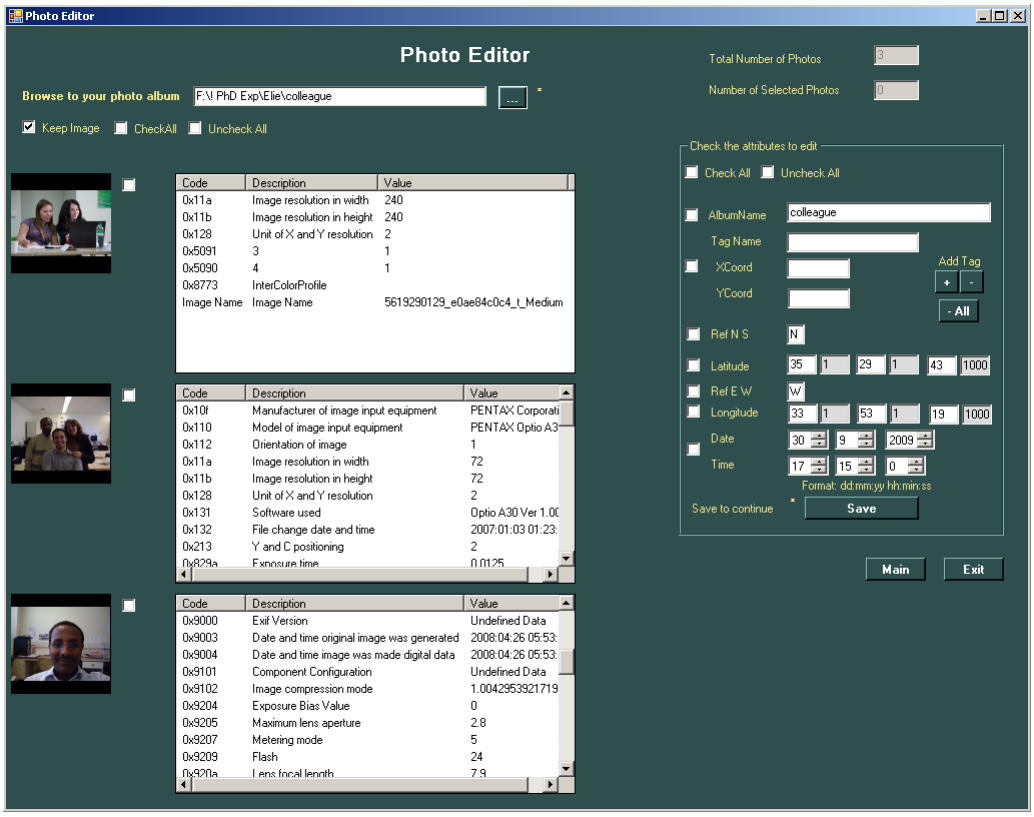


Figure 6.2: Screenshot of the photo editor module.

6.1.5 Rule Generator

The rule generator module, shown in Figure 6.4, offers to the main user the possibility to create the set of basic rules for the relationship discovery. Basic rules for social and co-referent relationships are generated separately as follows:

1. For social relationship discovery (Figure 6.4(a)): basic rules are generated in our prototype using a crowdsourcing scenario where photos are obtained from real-world dataset. The user must define the social relationship types to discover. Then, *RelTypeFinder* checks the validity of the chosen relationship. In addition, it verifies whether it is possible to retrieve from the content-sharing site at least 300 photos for each relationship type. After retrieving the set of photos, the main user can optionally edit the obtained photos (by removing detected objects that are not faces and tagging persons not depicted by the face detection algorithm [192]). As explained in Chapter 5 (Algorithm 1), rules are generated by extracting and combining explicit attributes such as photos' embedded information (time, date, location), photos' attributes (descriptions), and implicit information (number

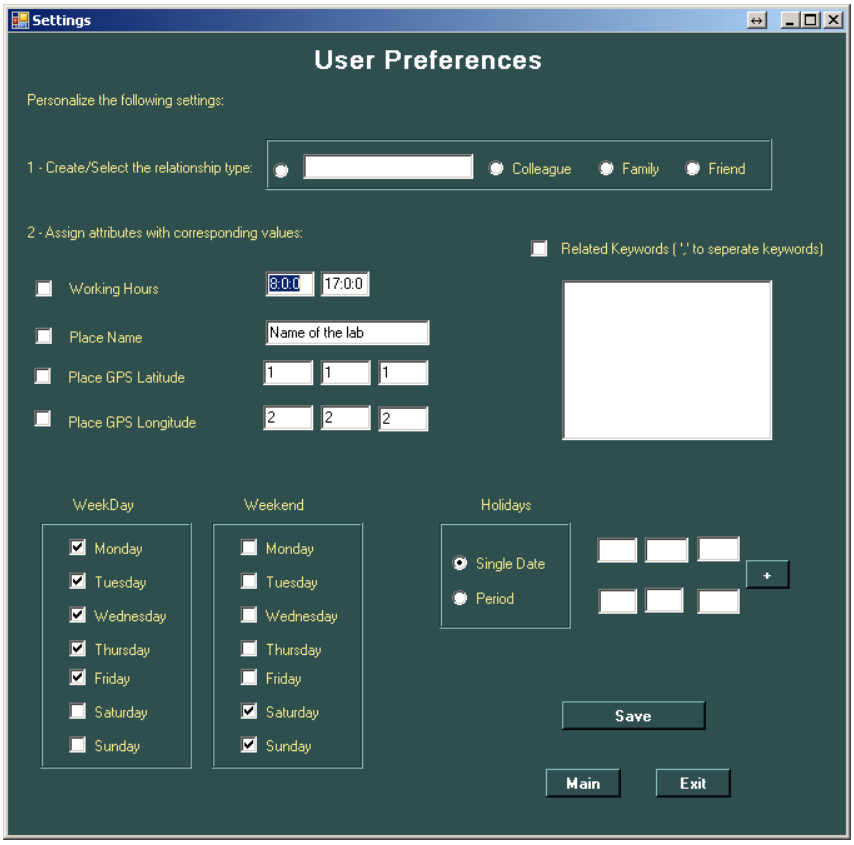


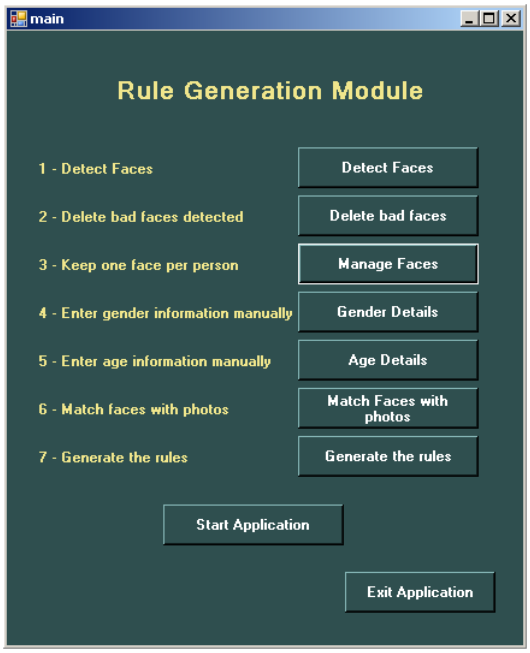
Figure 6.3: Screenshot of the user preference module.

of occurrences of persons together in photos, age and gender categories of persons who co-appeared together, age category of the persons compared to the age of the main user, etc.).

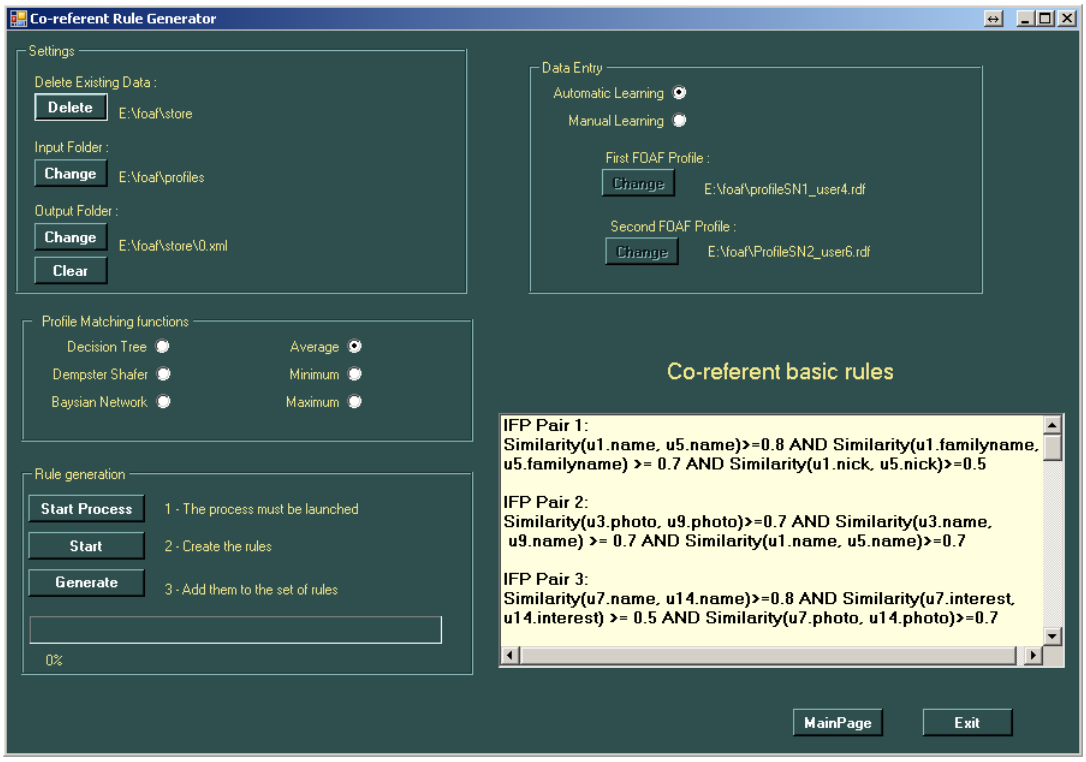
2. For co-referent relationship discovery (Figure 6.4(b)): basic rules are generated based on the characteristics of the extracted contacts' profiles from synthetic dataset. The synthetic dataset of profiles is obtained using the profile generator module. Using the rule generator module, one can generate the attribute-based basic rules and the profile-based basic rules (c.f. Chapter 5).

6.1.6 Rule Miner

This module extends the set of existing rules, generates new rules derived from the set of basic rules. Here, we implemented the Apriori algorithm [12] to mine the basic rules. The input of this module is the set of valid basic rules. We recall that valid basic rules are all the rules that, when



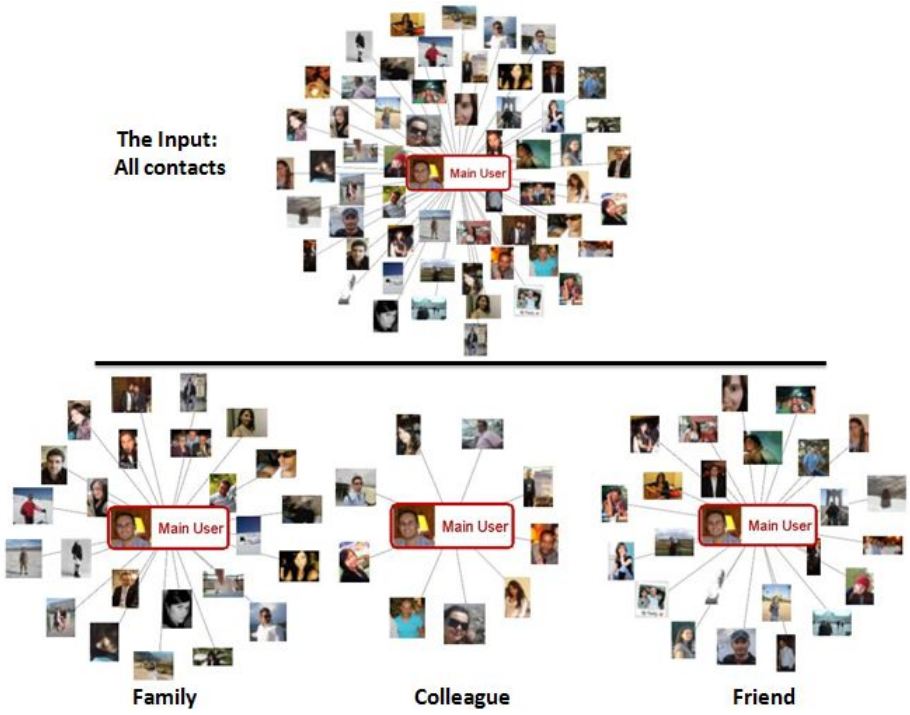
(a) Rules for social relationship discovery



(b) Rules for co-referent relationship discovery

Figure 6.4: Screenshot of the basic rule generator module for social relationships (a), and for co-referent relationships (b).

(a) Group of photos assigned to their corresponding relationship type



(b) Group of contacts displayed within their corresponding social star type

Figure 6.5: Screenshot of photos grouped by social relationship types (a), and of the star network with social relationship types (b).

applied on the profiles of a main user, match with at least one photo. Indeed, Apriori generates frequent itemsets within a given dataset where returned rules satisfy a minimum support and confidence threshold. Obtained rules are directly stored on the disk as RDF files. Figure 6.6 shows this module.

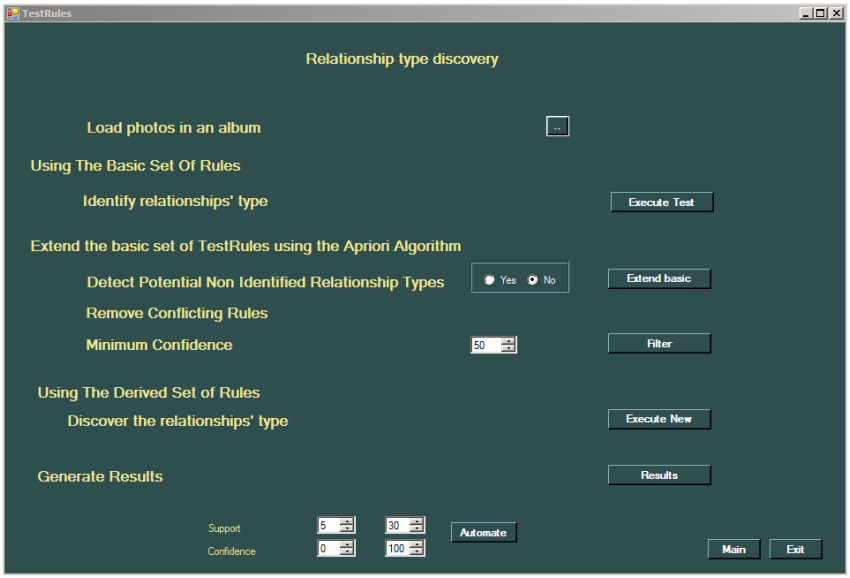


Figure 6.6: Screenshot of the rule miner module.

6.1.7 Relationship Finder

This is the main module of our prototype. It uses the set of basic rules to discover relationships' types and identify co-referent users. Then, using the set of derived rules, it aims to identify relationships yet undiscovered. We provide two screenshots (Figure 6.5(a) and Figure 6.5(b)) depicting two results of social relationship discovery. As for co-referent relationship discovery, it is shown in another screenshot (Figure 6.7).

In the next section, we start by presenting the dataset we used and collected before describing the set of experiments we conducted.

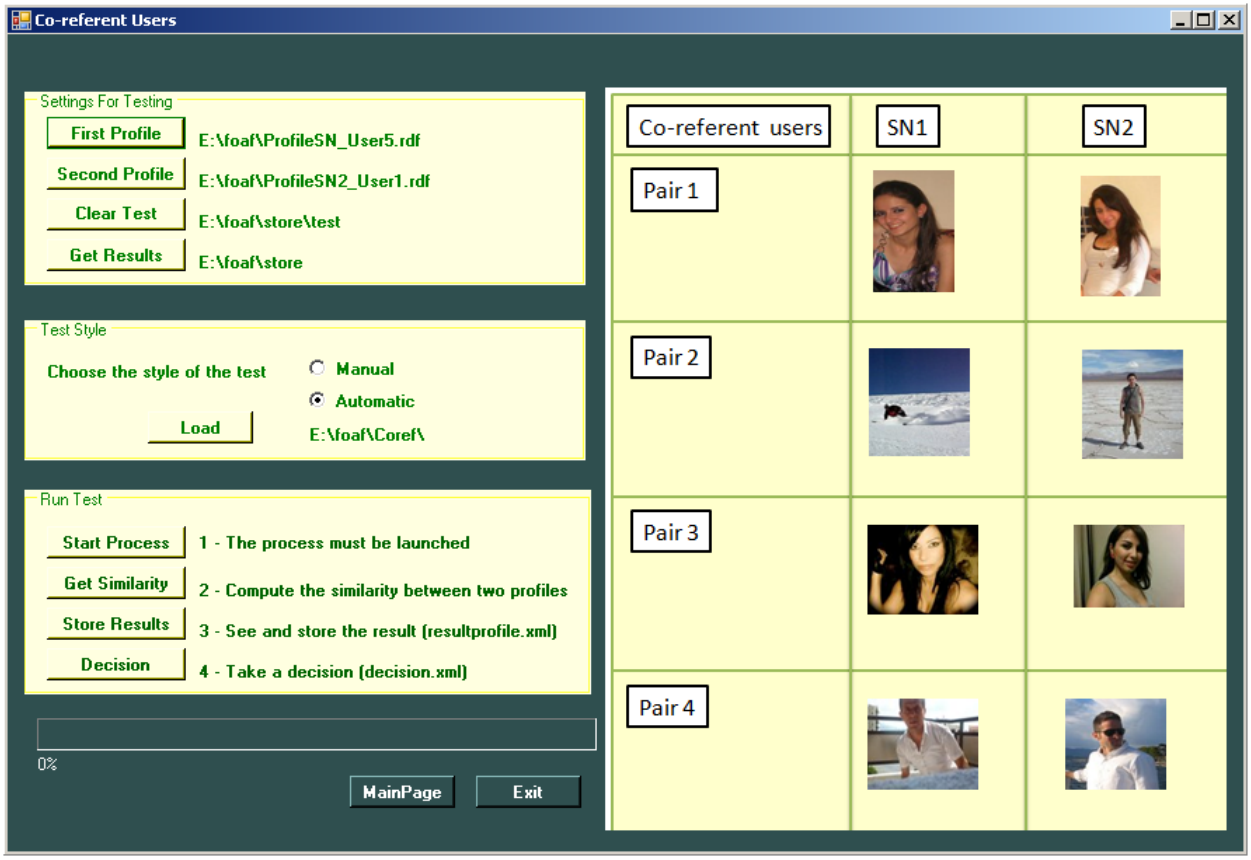


Figure 6.7: Screenshot of co-referent users identified.

6.2 Experimentations

In this section, we present the set of experiments conducted to demonstrate the efficiency of our approach. We first describe the datasets used and generated before presenting the evaluation of the experiments for the social and co-referent relationship discovery.

6.2.1 Datasets

Social network sites, while primarily designed for connecting users, provide a comprehensive interactive and photo-based environments that users can use to upload and share photos. As stated previously, in this work, we are interested in using all user profile's attributes including photos. Given the growing popularity of photos on social networks, admitting their capacity to model real-life events, and realizing their potential to capture natural and everyday human interactions, we strongly believe that real-world photos are required to test the efficiency of

our approach. On one hand, this is particularly true to identify social relationships between users. Therefore, as explained previously, we exploit various photos' attributes to identify social relationships. On the other hand, identifying co-referent users is not as intuitive as identifying social relationships mainly through the use of photos. Therefore, we mainly consider profiles' attributes and compute the similarity between profile photos. As a matter of fact, the use of real-world data is very important since it is difficult to simulate with synthetic data the content of users' profiles. However, it is easy to generate data by randomly varying attributes' values (e.g., typographical modifications, abbreviations, missing values, synonyms, etc.) when generating data by controlling the number of exact matches, missing attributes, incomplete values, etc.

It is worth noting that we encountered major difficulties while trying to collect real-world data that suits the need of our work. Within the context of social network sites, user profile and photo datasets are rarely made public due to several issues such as privacy, security, and availability. We invested a lot of time to find appropriate solutions in order to test our co-referent and social relationship discovery approaches. To handle this issue, we propose:

- For social relationship discovery: we collected a set of data from individual volunteers so to obtain real-world data that capture real-life events and personal interactions that are available on existing social network sites.
- For co-referent relationship discovery: we created synthetic datasets that represent profiles on two social network sites and controlled the generation of data.

Real-world and synthetic datasets are described in the following.

6.2.1.1 Real-world Datasets

We collected individual-related data from volunteers who accepted to give us access to collect the required information (namely FOAF profiles and photos) from their social network profiles. The dataset was collected from the social network profiles of 10 users. The average size of user' photos was 200 photos and each user had 100 contacts. Note that users were able to identify the appropriate social relationship type with each of their contacts. Here, we focus, as mentioned earlier, on three popular social relationship types: colleagues, relatives, and friends. Within the photos collected 25% are colleagues, 40% are family members, and 35% are friends. In our experiments, 22% of the photos contained one person, 50% contained two persons, and 28% contained more than two persons. The dataset consists of 2000 photos equally distributed between the 10 users. Table 6.1 summarizes the characteristics of this dataset. Note that users'

Table 6.1: Real-world dataset characteristics

Characteristic	Value
Number of photos	2000
Percentage of colleagues	25%
Percentage of relatives	40%
Percentage of friends	35%
Photos depicting one person	22%
Photos depicting two persons	50%
Photos depicting more than two persons	28%

profile information (name, age, gender) and photos' embedded metadata information (date, time, and GPS location) were available within the collected dataset. In addition, photos and photo albums have their attributes' values available (descriptions, comments, tags).

6.2.1.2 Synthetic Datasets

We used the profile generator of our *RelTypeFinder* prototype to generate profiles based on the following parameters:

- **Co-referent with same IFP:** profiles created with the same IFP value
- **Co-referent with different IFP:** profiles referring to the same real-world user but having different IFPs
- **Missing attributes:** number of common attributes between two similar profiles
- **Similar attributes values:** similarity between the randomly generated words with typographical errors, synonyms, abbreviations, incomplete values, etc.

The generation of these profiles takes into consideration the aim of two sets of experiments (Test 6-8, and Test 9) as described later on.

Evaluation Measures

To measure the relevance of our approach in detecting co-referent users and discovering correct social relationship types, we used three popular information retrieval measures [162]: *Precision* (P), *Recall* (R), and *F-score* (F). These measures are calculated as follows:

$$P = \frac{TP}{TP+FP} \quad R = \frac{TP}{TP+FN} \quad F = 2 \times \frac{P \times R}{P+R}$$

where TP, FP, and FN denote respectively True Positive (number of returned and correct matches), False Positive (number of returned but incorrect matches), and False Negative (number of not discovered but correct matches). These evaluation measures are illustrated in Figure 6.8.

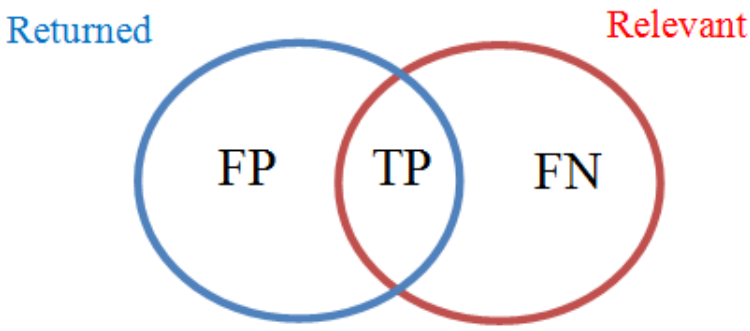


Figure 6.8: Evaluation Measures.

6.2.2 Relevance Of Our Approach

In the following, we describe different tests that we have conducted to evaluate the relevancy of our approach to identify 1) social relationships and 2) co-referent relationships.

6.2.2.1 Social Relationship Type Discovery Tests

In this set of experiments, we focused on the social relationship type discovery. The aim of this set of experiments was to evaluate: 1) the correctness of the generated basic rules, 2) the relevance and efficiency of the different rules of our proposed approach (basic, common sense, and derived rules), 3) the impact of varying the confidence and the support values on the results after mining, 4) the relevance of our proposed approach while varying the percentage of annotated photos, and 5) the benefits to our approach when applied on a partially labeled star social network.

Test 1: Rule Correctness

The aim of this test was to validate the correctness of our set of social basic and common sense rules. The evaluation of the correctness of the basic rules generated by our proposed social rule generation methodology consisted of:

- **Test 1.A - A subjective survey:** the aim of this qualitative user study was to get the feedback of a set of participants to test the correctness of the set of basic and common sense rules using a survey that we prepared for this purpose. Their mission was to answer questions that reproduce different real-life situations depicted in photos in order to validate the type of relationship implied from the photos. The participants were 120 graduate students in computer science that we selected by verifying that they had good knowledge related to current social network sites. These 120 students were in different computer science fields: 40 multimedia students, 40 systems and networks students, and 40 telecommunication students. They were active social network users on at least one social network site. The participants were connected with different persons within their contacts, including colleagues, relatives, and friends. They had also shared with their contacts a number of photo albums.

- **Test 1.B - A comparative study:** we compared our set of basic rules with the set of rules presented in [167]. In fact, we applied both sets of rules on the same real-world dataset (described earlier). We compared the obtained results in terms of F-score results.

In the following, we detail these two studies.

Test 1.A - Survey structure

The participants were asked to fill in an online questionnaire[1] to evaluate the set of basic and common sense rules. The survey was divided into three parts:

- **Part 1:** this part included some personal questions to measure how familiar participants were with the use of social network sites.

- **Part 2:** this part included questions to get the feedback, according to each participant's personal experience, to evaluate when and in which context persons appearing in a photo are assigned to a specific relationship type. Participants were asked to answer questions that describe different real-life situations that depict persons in photos. Participants had to choose the most appropriate relationship type according to their common sense and social network experience. Users had the possibility to choose more than one social relationship type (given three options: colleagues, relatives, and friends), or to fill a new relationship if needed. The aim of this part was to compare the obtained answers on this set of questions. Note that these questions were in fact the body of the basic rules and the expected answer of the participants was the head of the rule (the relationship type).

[1] http://sigappfr.acm.org/rulessurvey/

- **Part 3:** this part aimed to validate the set of common sense rules that we manually generated. Recall that this set of rules is used to extend the set of basic rules based on our common sense. Similarly to Part 2 of the survey, the participants were given the body of the rules that describe different real-life situations. In return, the participants had to rate how likely they agree on the given common sense rules according to their perceived common sense.

We evaluated and compared the results obtained from Part 2 and Part 3 of the survey (Part 1 was only mainly targeted to verify that the participants had enough knowledge on social networks to participate in the survey).

Test 1.A - Survey Results - Part 2:

Overall, this evaluation revealed that the majority of the participants approved the rules generated for each relationship type. In fact, each question in the Part 2 of the survey corresponded to one rule that was part of our set of basic rules listed in Chapter 5. We compared the given answers from the participants and the relationship types (head of rules) already obtained using our rule generation methodology. In the following, we evaluated the percentage of answered relationships that matched with the head of the set of basic rules. Figure 6.9 illustrates the results as described below:

- **Agree:** over 80% of the surveyed participants validated that the set of basic rules is able to discover the corresponding social relationship types (stated in the head of the rules). Precisely, 80% for colleagues-related rules, 81% for the relatives-related rules, and 86% for the friends-related rules.
- **Agree but add another relationship(s):** of the surveyed participants, an interval ranging from 14% till 21% reported that they agree on the given rules but consider that these rules could also be used to discover other relationship type(s). Precisely, 21% considered that colleagues' basic rules may also refer to other relationships, 17% considered the same for relatives' basic rules, and 14% for friends' basic rules.
- **Don't agree:** a small percentage varying between 12% to 17% didn't agree on the rules: 13% felt that these basic rules were not able to discover colleagues' relationship types, some 17% of the participants considered that relatives-related basic rules were not suitable, and 12% of the participants didn't agree on the friends-related basic rules.

Recall that participants were able to choose more than one relationship type per question in the Part 2 of the survey (this explains the scale of the y-axis in Figure 6.9).

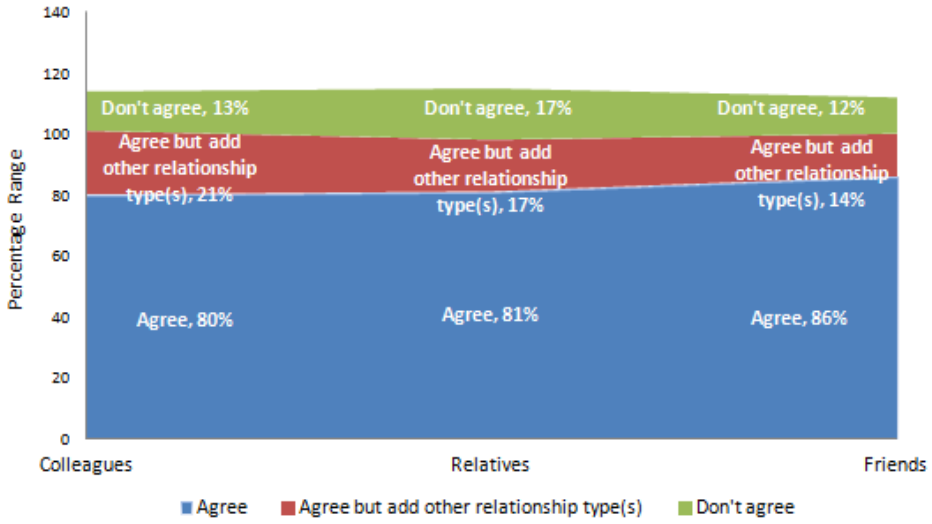


Figure 6.9: Basic rules evaluation results for colleagues, relatives, and friends relationship types.

Test 1.A - Survey Results - Part 3:

Similarly, the participants were given another set of rules, the set of common sense rules. They were asked to rate how much likely they find this set of rules' statements true according to their perceived common sense. The results are shown in Figure 6.10. The obtained results from this evaluation confirmed that the generated rules and the common sense rules hold true according to the participants.

Test 1.B - Comparative Study

In this test, we compared our set of basic rules with the set of rules used in [167]. In fact,

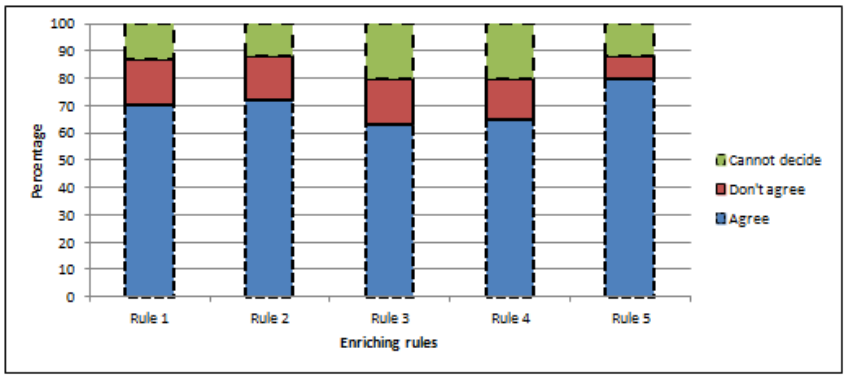


Figure 6.10: Common sense rules evaluation results.

the work described in [167] is close to our work. In addition, the authors provided us with the set of their rules. As stated earlier, both sets of rules were tested on our real-world dataset. However, we had to disable the rules related to the colleagues relationships from our set of basic rules (colleagues, relatives, and friends) since the rules provided in [167] aimed to discover two categories of relationship types: relatives and friends (acquaintances). We conducted this test using only basic rules without mining. Table 6.2 summarizes the F-scores results obtained.

	Relatives	**Friends**
Singla et al. [167]	58.46 %	60%
Our approach	73.87%	75.5%

Table 6.2: F-score results when comparing the basic rules

As observed in Table 6.2, for both relationship types (relatives and friends), our approach achieved a higher F-score value than the one of the other approach. Indeed, while the rules in [167] are limited to a subset of attributes (age, gender, co-appearance), our rules have higher expressiveness since they consider a wider set of attributes extracted from photos such as date, time, location, etc.

Test 2: Efficiency of different rules: basic, common sense, and derived

This test aimed at showing to which extent our approach was able to discover the correct relationship types between a user and her contacts. We considered for this test the real-world dataset of photos. We randomly removed 50% of the photos' attributes and their embedded metadata.

1. **Test 2.A - Using the basic rules only:** only the set of basic rules is used. The basic rules that we collected using our crowdsourcing scenario are applied.

2. **Test 2.B - Using the basic and common sense rules:** the common sense and basic rules are both applied. Recall that in this work we consider the worst case scenario where no relationship types were initially available. Consequently, the set of basic rules was first applied to discover some relationship types.

3. **Test 2.C - Using the derived rules:** as mentioned earlier, in order to extend the set of basic rules, we used the Apriori algorithm which is able to generate extended rules that satisfy a given score (minimum support and confidence threshold scores). Here, the support score refers to the percentage of photos that satisfy the rules. The confidence score refers to the fraction of photos containing the body (antecedent) of a rule and that also contain the head (consequent). While, in general, high support and high confidence

scores are desirable in different applications (e.g., Market Basket Analysis), in our work low support scores were also important since they allowed to discover new relationship types useful for the discovery task. We therefore executed the Apriori algorithm with a minimum support score of 1% while we varied the value of the minimum confidence score.

Table 6.3: Precision and Recall values with different confidence scores

	Colleagues		Relatives		Friends	
	P	R	P	R	P	R
Test 2.A - Using the Basic Rules Only	53%	60%	55%	65%	54%	54%
Test 2.B - Using the Basic and Common Sense Rules	58%	60%	60%	65%	58%	58%
Test 2.C - Using the Derived Rules, Confidence 0%	60%	85%	65%	76%	70%	70%
Test 2.C - Using the Derived Rules, Confidence 50%	65%	85%	55%	65%	74%	77%
Test 2.C - Using the Derived Rules, Confidence 100%	58%	60%	60%	65%	58%	58%

The obtained results are shown in Table 6.3. In the following, we discuss these obtained results:

1. **Test 2.A:** measuring the precision and recall of relationship discovery for basic rules is important; however, since basic rules don't require any mining in *Test 2.A*, their precision and recall values are unchanged (they don't change in function of confidence scores). The obtained results show that our basic rules are able to detect correctly a part of social relationships which are already known by the profiles' owners.

2. **Test 2.B:** similarly to the basic rules, the common sense rules don't vary in function of the confidence scores. The results of *Test 2.B* clearly show an improvement in the precision results for the three relationship types. The percentage of correctly identified relationships increased between *Test 2.A* and *Test 2.B* from 53% to 58% (colleagues), 55% to 60% (relatives), and 54% to 58% (friends). As for the recall values, they remained unchanged for the colleagues and relatives relationships, whereas the recall score for relatives increased from 54% to 58%.

3. **Test2.C:** we detail three main points:

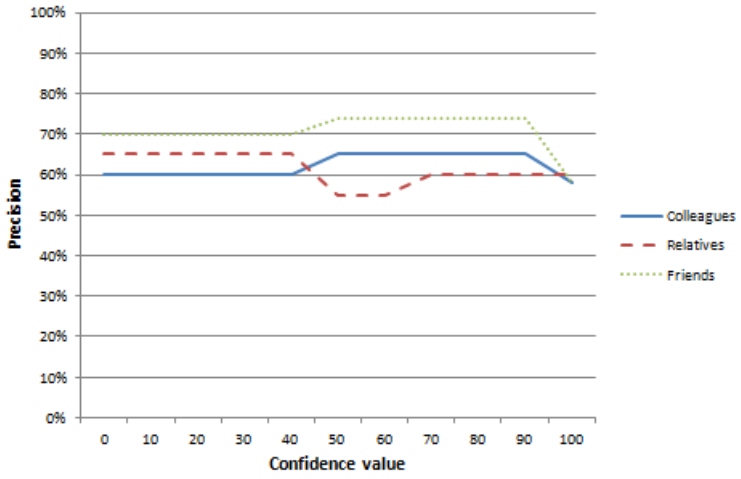
   (a) **Confidence score of 0%:** A minimum confidence of 0% means that all the mined rules obtained from the Apriori algorithm were used. However, since all the obtained derived rules were applied, this means that the system might also have returned false positive results. Not only the number of true positive increased, but also the number of all returned results. Consequently, this had a direct influence on the precision measure.

(b) **Confidence score of 50%:** only rules with a confidence score above 50% were taken into consideration. This filtered out the initial set of rules and gave the following result: the precision value, for the colleagues and friends relationship types, increased respectively of 5% and 4%. For the relatives relationships type, the precision dropped down. Although rules with confidence score less than 50% were capable to identify a number of relatives relationships, they were eliminated when the confidence score was higher than 50%. So, a number of relatives relationship type previously identified with 0% confidence score was not identified with 50% confidence score. As a proof to this interpretation, we can see that the recall score dropped down for the relatives with a minimum confidence of 50%. This means that some true positive results were not identified. In the cases of colleagues and friends relationship types, the increase of the recall percentage was due to a decrease in the number of the returned false positive results.

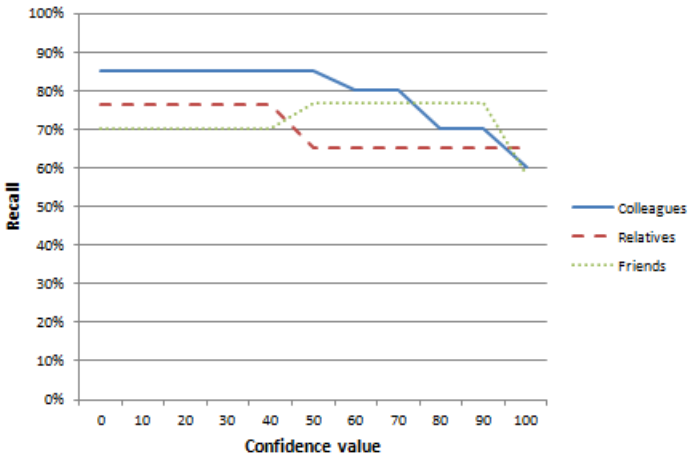
(c) **Confidence score of 100%:** only rules, which had a confidence of 100%, were selected. This was the uppermost value that could be applied. The obtained sets of mined rules were very similar to the set of basic rules since all derived rules, except the set of basic rules, were pruned since their confidence scores were below 100%. As a result, similar precision and recall values were achieved as if mining was not applied as shown in Table 6.3.

Figure 6.11 shows the results obtained in *Test 2.C* (using the derived rules) while varying the value of the confidence score. One can see that the precision and the recall scores varied in different manners for each considered social relationship type. Indeed, the interpretation of these results must jointly consider the two above-mentioned measures. In addition, we wanted to evaluate the impact of common sense rules on the results obtained using the derived rules. Table 6.4 shows that the Precision and Recall values while using the common sense rules (*Test 2.C, Confidence 0%*) outperformed slightly the results when these rules were not used (*Test 2.C, Confidence 0%*). Note that similar results were obtained for different confidence scores. In fact, such results were expected and they validate the choice of using rules to propagate knowledge.

Obviously, these results confirm the capacity of our approach to discover relationships between a main user and her contacts in the context of social networks. In addition, applying derived rules, as described in our approach, returned satisfactory results. It is true that the derived rules (*Test 2.C*, Confidence 0% and 50%) performed better than the basic rules and common sense rules (*Test 2.A* and *Test 2.B*), or achieved similar results (*Test 2.C*, Confidence 100%), over the collected dataset. However, it is important to recall that these derived rules are



(a) Precision scores



(b) Recall scores

Figure 6.11: Results obtained when varying the values of confidence scores for the precision scores (a) and recall scores (b).

Table 6.4: Impact of common sense rules on precision and recall values - Confidence 0%

	Colleagues		Relatives		Friends	
	P	R	P	R	P	R
Test 2.C - Derived rules: using the basic and common sense rules	60%	85%	65%	76%	70%	70%
Test 2.D - Derived rules: using the basic rules	55%	60%	57%	65%	70%	56%

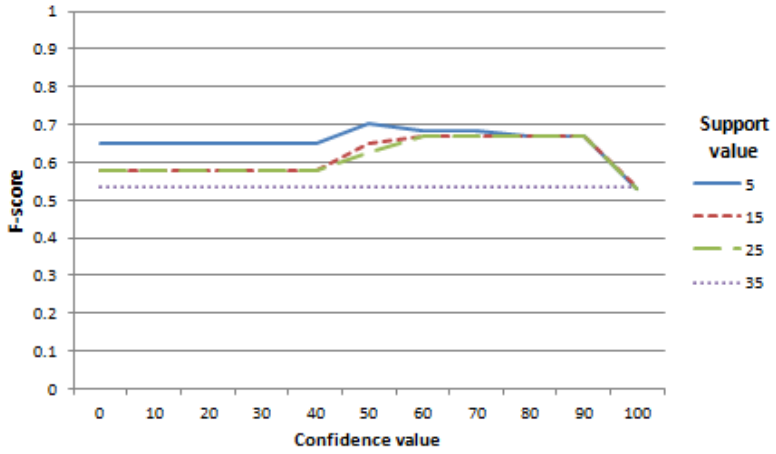


Figure 6.12: Comparing the F-score values while varying the support and the confidence values of the mined rules.

extracted from the set of basic rules. That means that basic and common sense rules must be available and applied over the collected dataset so that the derived rules can be generated for the purpose to discover relationship types, yet unlabeled.

Test 3: Impact of varying the confidence and the support scores

The aim of this test was to evaluate the impact of the confidence and the support scores on the obtained results. Through this test, we investigated the accuracy of our approach while varying support and confidence scores of the mined rules. The F-score values for all of the three social relationships (colleagues, relatives, and friends) were computed. Results of the three relationships showed the same pattern. The support score was varied from 0% to 100%. Support scores with a percentage greater than 35% gave the same results, and consequently we don't show them on the result obtained in Figure 6.12.

Based on the computed results shown in Figure 6.12, we were able to pin down two main

observations:

1. **When the confidence value was equal to 100%:** for different support values, regardless of the F-score value, all the tests reported the same F-score value. In such cases, the derived rules were filtered out and eliminated from the new set of rules. Consequently, the set of rules was reduced to the set of basic rules available before applying the mining algorithm.

2. **The lower the support value, the higher the F-score value:** rules with low support value use a wider set of rules to discover relationships, thus they achieved greater F-score values than rules with higher support values.

Test 4: Relevance of our proposed approach while varying the percentage of photos' attributes and their related embedded information

The aim of this test was to study the effect of available photos' characteristics on discovering the social relationship types using our approach. To do that, we conducted four experiments with different available photos' characteristics percentages. We chose to vary the photos' characteristics percentage from 25% to 100%. The confidence value varied from 0% to 100% in each of these experiments. In this test, we also varied the percentage of available metadata and attributes available in photos and photo albums. Photos' characteristics, composed of embedded metadata information and photos' related attributes, were randomly removed or differently assigned empty values. The obtained results in terms of F-score for colleagues, relatives, and friends relationships are provided in Figure 6.13.

One can easily observe that the more available photos' characteristics within the dataset, the more the result is accurate. This observation was true for all the categories and for all the confidence values. Another important observation to note is related to the variation of F-score value which drops down when reaching 90% of confidence value. This is due to the limited number of rules with a confidence value between 90% and 100%. In other terms, since the set of used rules was restricted to the basic rules, the obtained F-score value was similar to the one without mining. This experiment showed also that for a 100% set of annotated photos, a high F-score value was obtained for:

1. Colleagues relationship: between 80% and 100%,

2. Relatives relationship: between 90% and 100%,

3. Friends relationship: between 75% and 80%.

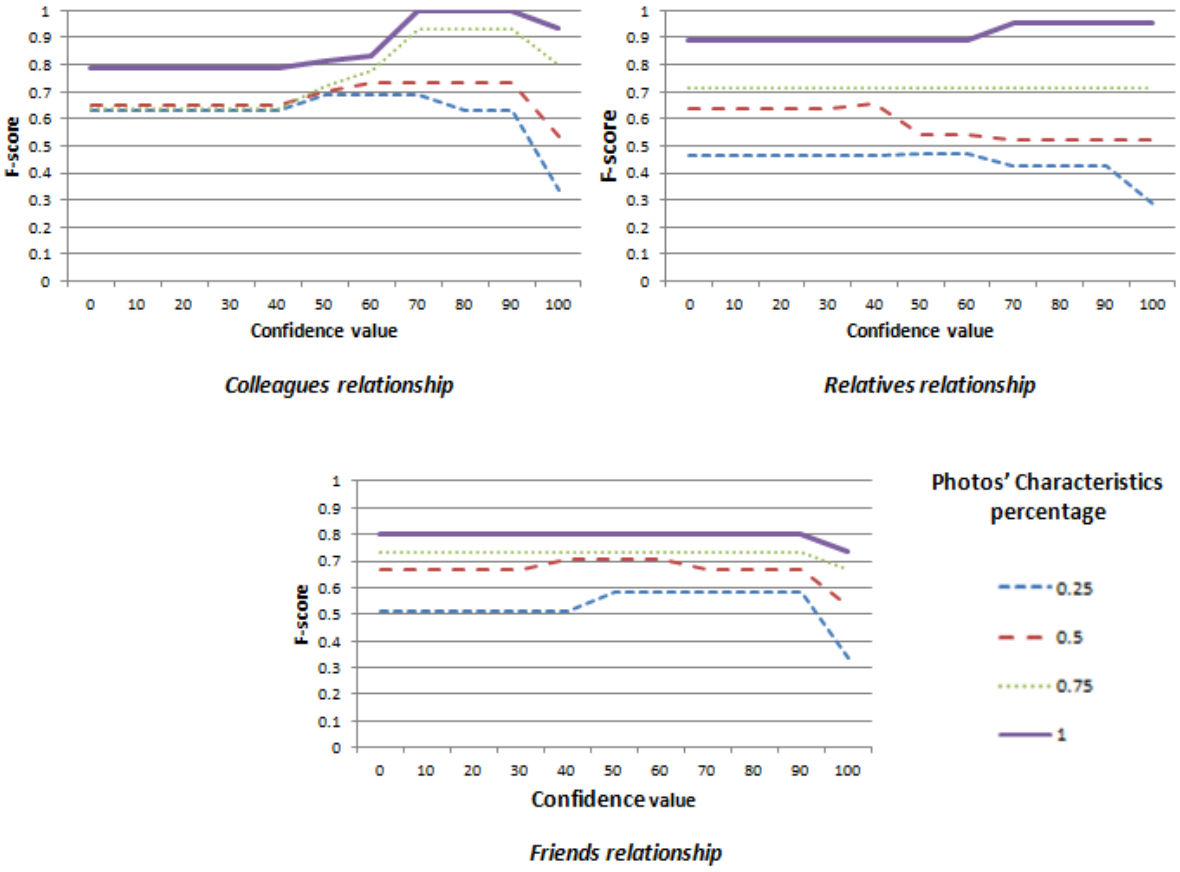


Figure 6.13: Comparing the results of F-score while varying the percentage of photos' characteristics for each relationship type.

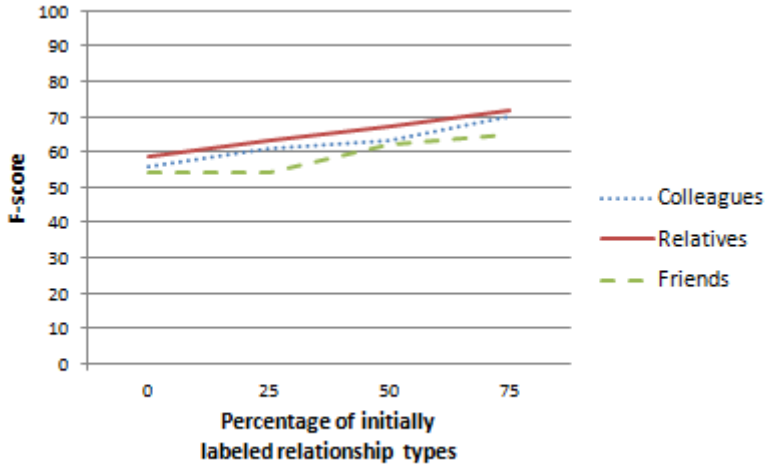
For a percentage of 25% of available photos' characteristics (this was the lowest percentage), and while varying the confidence value between 0% and 90%, the F-score value achieved a score between 61% and 69% for colleagues relationship, between 41% and 49% for relatives relationship, and between 51% and 59% for friends relationship. This means that accurate and efficient results can be obtained even with a set of photos having its photos' characteristics partly available.

Test 5: Benefits of partially labeled star social networks

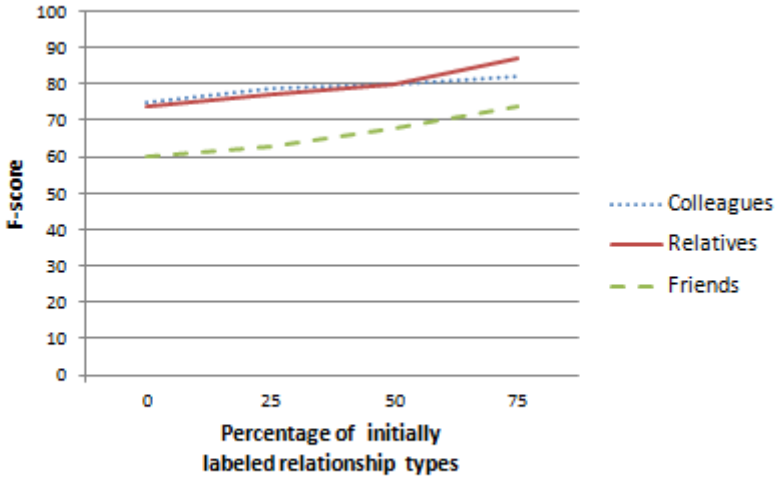
In this test, we asked the 10 volunteers, owners of the collected real-world dataset as explained earlier, to randomly identify (appropriately) some social relationship types with their contacts. The aim of this test was to study the effect of having a partially and correctly labeled star social networks. In fact, in the previous tests (Test 1, Test 2, Test 3, and Test 4), we evaluated our approach without having any known social relationship type between a main user and her contacts. However, we were interested in checking till which extent available relationship types can be beneficial for our approach in terms of efficiency. To that end, we varied the percentage of initially labeled relationships given by the main users from 0% to 75%. Note that for a percentage of 100% labeled relationships, the relationship discovery is not needed. We tested our approach using different percentages of labeled relationship types. Figure 6.14(a) and Figure 6.14(b) summarize the obtained results.

In Figure 6.14(a), basic rules performance was studied along the variation of the percentage of labeled relationships: the results reported a higher F-score value when the percentage of labeled relationship increased. Similarly, in Figure 6.14(b), the derived rules were used and similar results were found. It should be noted that in Figure 6.14(a), the F-score value stayed unchanged for 0% and 25% of initially labeled relationship types. Since main users randomly labeled relationship types, this explains why no increase was observed in the F-score value for these two percentages (0% and 25%). Indeed, we consider that the main users know their contacts so labels are appropriate and match with the ground truth. Consequently, the number of false positive returned results didn't change.

In another test, illustrated in Figure 6.15, we decided to focus on one relationship type in order to evaluate the results using different types of rules. To do so, we chose the relatives relationship and we compared the results while varying the percentage of initially labeled relationships. Figure 6.15 highlights the results' improvement when the percentage of initially available labeled relationships was higher. It is to be noted that for derived rules with a percentage of 75% the best results were obtained, whereas the lowest results were obtained in two cases: 1) only using basic rules, and 2) when using derived rules with a percentage of 100% (the

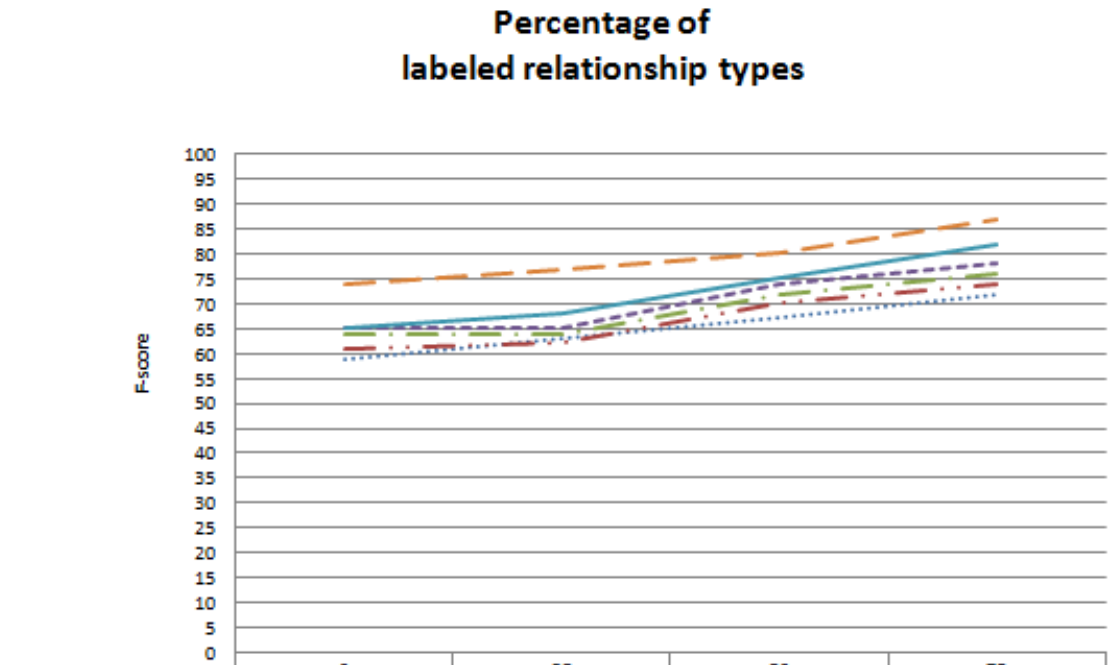


(a) Using only the set of basic rules



(b) Using the set of derived rules with a confidence score of 75%

Figure 6.14: Results obtained when varying the values of initially labeled relationship types when using only the set of basic rules (a) and when using the set of derived rules with a confidence score of 75% (b).



	0	25	50	75
Basic Rules	59	63	67	72
Enriching Rules	61	62	70	74
Derived Rules 0%	64	64	72	76
Derived Rules 25%	65	65	74	78
Derived Rules 50%	65	68	75	82
Derived Rules 75%	74	77	80	87

Figure 6.15: Detailed results related to the relatives relationship and tested using different types of rules. The results were obtained by varying the values of initially labeled relationship types.

obtained rules were very similar to the set of basic rules).

The results of this test showed that initially labeling relationship types is of considerable importance for the relationship type discovery. While results were globally improved for basic and common sense rules, the initially labeled relationships were of valuable importance for the derived rules. In fact, they facilitated the selection of the most relevant information to be used by the mining algorithm. Consequently, this brought the greatest benefits to generate more reliable derived rules as it had been reported within the results of this test. In reality, this feature can be exploited when some manually annotated relationship types exist within the star social network of users.

To sum up, the previously described tests (Test 1-5) underline significant relationship identification possibilities using the basic rules alone. Furthermore, while considering also the set of common rules we observed an improvement in the results. This improvement was confirmed when the derived rules were applied, particulary with low confidence values. Another important point to note is that our approach reported better results when the star social network is partially labeled. In the worst case, when no relationships are available, the set of basic rules proved their efficiency to handle such issue. Finally, using photos' characteristics is of considerable importance, thus exploiting other attributes related to photos is a promising dimension to exploit in the future.

6.2.2.2 Co-referent Relationship Type Discovery Tests

Although an IFP (such as the email address) can be chosen within a single social network, its usage across different social networks is much more intriguing (c.f. Chapter 5). However, a varying subset of co-referent users can still be identified using the IFP. In fact, it fails to detect the remaining part of co-referent users. Therefore, we propose to exploit the set of detected co-referent users using the IFP to create a set of basic rules (attribute-based and profile matching rules) that are trustworthy to identify a larger number of co-referent users across different social networks.

We evaluated the results against the generated ground truth. To this end, we generated synthetic dataset, as described in Table 6.5, with 400 generated profiles. Only 20% of the generated profiles refer to the same real-world persons but having different IFP (the email address) values: this is our generated ground truth. The accuracy to identify co-referent users was tested by varying the number of common attributes between pairs of profiles. In fact, we tried to simulate the real-world profiles of co-referent users by creating profiles referring to same

Table 6.5: Co-referent datasets characteristics

Characteristic	Value Test 6-8	Value Test 9
Number of profiles	400	1000
Number of profiles' attributes	15	10
Percentage of co-referent users	20%	30%

real-world persons whose profiles are often not exactly similar.

The aim of this set of experiments was to evaluate: 1) the relevance of our basic rules to identify co-referent users, 2) the benefit of combining the set of basic rules and the IFP-based method, 3) the relevance of using the derived rules for co-referent relationship type discovery, and 4) the efficiency of using the blocking technique to optimize the obtained results.

Test 6: Relevance of basic rules to identify co-referent users

In this experiment, the goal was to test the effectiveness of our set of basic rules in the absence of a global IFP identifier. We conducted this experiment in order to identify co-referent profiles using the created set of co-referent basic rules which is composed of attribute-based and profile-based matching rules. The accuracy of these two methods is shown in Figure 6.16.

As the results show, the set of basic rules can achieve over 90% accuracy using the F-score measure. However, the F-score value was low (less than 30%) when the compared pairs of profiles had over 60% of their attributes' value different. The F-score accuracy increased as the number of different attributes' value decreased. Starting with a difference of 40% between the pairs of profiles, the F-score accuracy increased exponentially to reach a value of 98% when the pairs of profiles had exactly the same attributes' values.

This means that our set of basic rules wasn't reliable to identify co-referent users for profiles having a high number of different attributes (over 70%). However, their perceived usefulness is effective when dealing with absence of a global IFP, particularly when the similarity of attributes' value was over 60%.

Test 7: Benefits of combining IFP-based method and the set of basic rules

The aim of this experiment was to evaluate the benefit of combining the use of our set of basic rules and the IFP to identify co-referent users. This means that, co-referent pairs of profiles are identified using the IFP, then the basic rules are used to identify the remaining co-referent profiles yet undiscovered. We compared the obtained results when: 1) using only IFP attribute, and 2) combining the use of basic rules and IFP. Figure 6.17 shows the results of this test.

Based on the results shown in Figure 6.17, we were able to pin down the following observa-

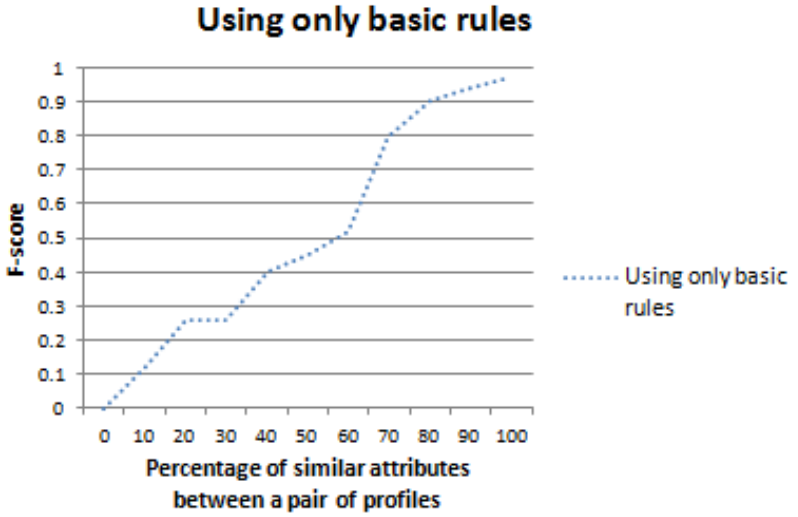


Figure 6.16: Comparing the results of F-score while varying the percentage of attributes having similar values

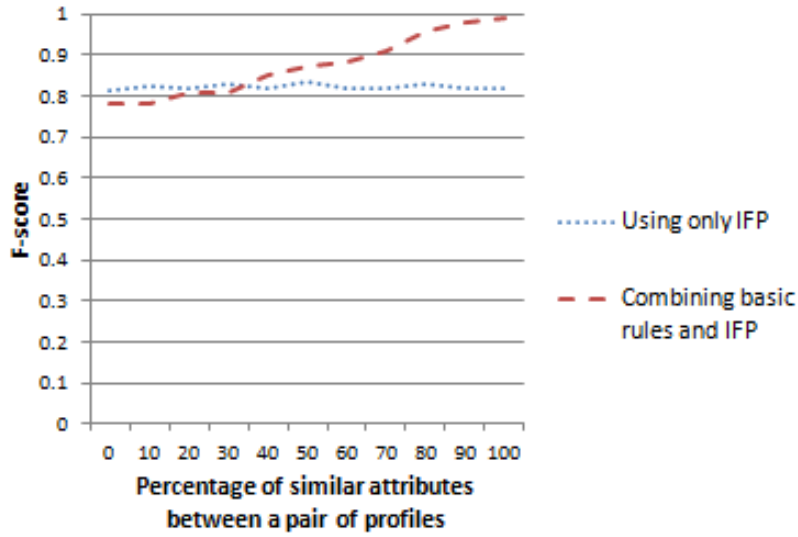


Figure 6.17: Comparing the results of F-score while varying the percentage of attributes having similar values.

tions:

- Although a small number of co-referent users was not detected, we were able to identify most of the co-referent users part of our ground truth (co-referent users with different IFP attributes). The results were encouraging overall.
- While the IFP-based method alone was not affected by the variation of the percentage of attributes, the combination of basic rules and IFP evolved proportionally in function of the percentage of these attributes.
- Starting from a percentage of 25%, which corresponds to profiles where only 20% of their attributes had similar values, the combination of basic rules and IFP achieved better results than the IFP-based method. This trend was accelerated when the percentage of attributes with similar values was above 50%. Accordingly, it would be more advantageous to apply our proposed method, except when less than 25% of the attributes are common. In such case, the high number of false positive matches affected negatively the F-score results.
- As the number of detected profiles decreased, the percentage of attributes with different values increased. The highest number of correctly detected profiles corresponded to profiles with a low percentage of attributes of different values. The lowest number of detected profiles corresponded to profiles that had a high percentage of attributes with different values (the results were very similar to the ones yielded by the IFP-based method).

Test 8: Relevance of our co-referent rules approach using derived rules

In this test, we aimed to show till which extent the derived rules were able to correctly identify co-referent users. As in the previous experiments, we varied the percentage of common attributes' values among compared pairs of profiles. We measured the F-score value while using the set of basic rules, then we used the extended set of rules to measure the F-score value with different confidence scores. The obtained results are reported in Table 6.6.

Table 6.6: F-score values with different confidence scores

	0	20	40	60	80	90	100
Test 1 - Basic Rules Only	0%	26%	40%	52%	90%	94%	97%
Test 2 - Derived Rules, Confidence 0%	0%	26%	37%	55%	90%	92%	97%
Test 2 - Derived Rules, Confidence 50%	0%	26%	40%	70%	94%	97%	97%
Test 2 - Derived Rules, Confidence 100%	0%	26%	37%	55%	90%	92%	97%

In line with the previous results in this chapter, results provided in Table 6.6 reported higher F-score values when using derived rules than those achieved by the set of basic rules.

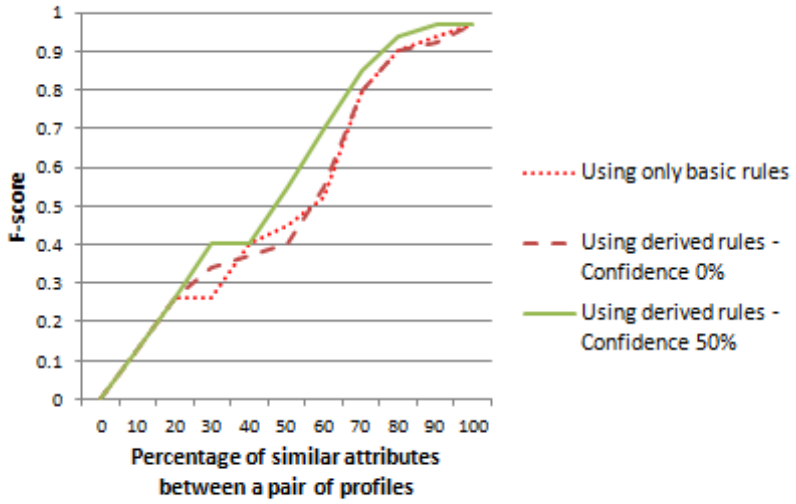


Figure 6.18: Comparing the results of F-score while varying the percentage of attributes having similar values.

Particularly, for a confidence score of 50%, the derived rules yielded the best F-score value as shown in Figure 6.18. Generally speaking, the F-score value increased linearly in terms of accuracy. This underlines the linear dependency to the number of common attributes' value of the compared profiles.

Test 9: Efficiency of using the blocking technique to identify co-referent users

In this test, we evaluated the efficiency when the blocking technique was used to identify co-referent contacts. Blocking techniques consist at reducing the set of candidate co-referent users by removing pairs of candidates that probably cannot match. Therefore, in this test we search for co-referent users among blocks based on the relationship type. Using a different dataset from the previous tests (Test 6-8), we conducted a series of experiments while varying the number of co-referent contacts of the ground truth. Actually, tests were carried out on a manually designed dataset where we collected 10 users' profiles on two different social sites. Each user had 100 contacts on each site. Each profile had at least 10 common attributes. To build our ground truth, we retrieved from each users' contacts those having common email addresses (considered as the IFP). Table 6.5 describes this dataset. We therefore conducted the tests by excluding the email address to compare the results of our profile matching approach with and without applying the blocking technique.

Having this in mind, we also studied the effects of persons having 0% common contacts

among their two profiles and we increased this percentage to the maximum of common contacts in our dataset which is 30%. The F-score results obtained are shown in Figure 6.19:

- One can easily observe that the results achieved while using the blocking technique for profile matching were the most accurate results. This observation was true along all the percentages of co-referent contacts.
- Another important observation to note was related to the variation of F-score value which was almost the same for both methods for a percentage less than 5%. Thus, there was no additional benefit regarding the accuracy of the use of the blocking technique for profile matching if the percentage of common contacts among two profiles was less than 5%. This was due to the limited number of profiles to identify.
- As the number of co-referent contacts increases between two profiles, it is worthwhile to use the blocking technique. The interpretation of these results must take into consideration the number of false positive results returned in function with the number of true positive results to detect. Whereas the number of false positive was inversely proportional to the results, the number of true positive was proportional to the obtained results. Consequently, since the number of false positive was almost constant, the F-score increased when the number of true positive results also increased.

As a result, using the blocking technique for profile matching achieved better results. Altogether, these results showed that using the blocking technique :

1. Reduced the number of returned false positive results
2. Increased the number of returned true positive results
3. Excluded a number of false positive matches thanks to co-referent profiles found by comparing block of the same relationship type.

To summarize, the previously described tests (Test 6-9) proved the effectiveness of the basic rules to detect co-referent users. Combining these basic rules with an IFP reported a significant capacity to detect further co-referent profiles. Similarly to the results obtained to discover social relationships, derived rules were also of valuable utility for finding co-referent users. As commonly known identifying co-referent users can be prohibitively expensive, therefore we validated that using a blocking optimization technique while taking into account the social relationship types of users can enhance the obtained results. Finally, using all profile's attributes as detailed

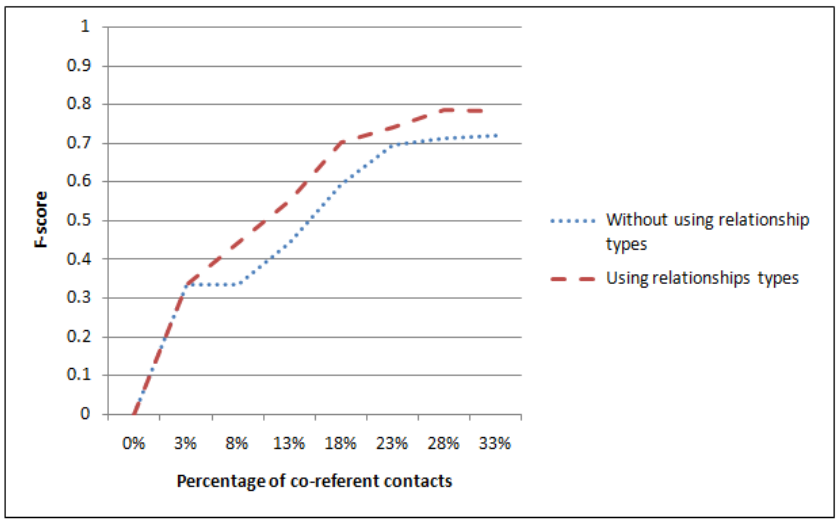


Figure 6.19: Comparing the results of F-score with and without using the relationship types.

in our approach (c.f. Chapter 5) showed an interesting reliability to detect co-referent users. Exploring jointly photos characteristics for generating and identifying co-referent users is another promising direction.

6.2.3 Time Analysis

We experimentally analyzed the execution time complexity of our relationship discovery algorithm for social and co-referent relationship discovery. We varied the number of user profiles (for co-referent relationship), and the number of photos (for the social relationship) included in the test. All experiments were carried out on a 2.8 GHz Intel Centrino machine 4GB RAM. We measured the execution time before and after applying the rule mining.

Test 10: Time Analysis for social relationship discovery

The aim of this test was to measure the temporal aspect while 1) varying the confidence score values and 2) increasing the number of photos. We used different album photo collections ranging from 100 to 500 photos per album and per user. It was obvious that without applying the rule mining, the overall time to scan and process the collection of photos was higher than the time needed to process the remaining photos after mining. As mentioned previously, the rules obtained after mining were only applied to the remaining photos that the basic set of rules haven't discovered previously. In Figure 6.20, we show the time execution after mining with different support and confidence values. We varied the confidence value that is represented on

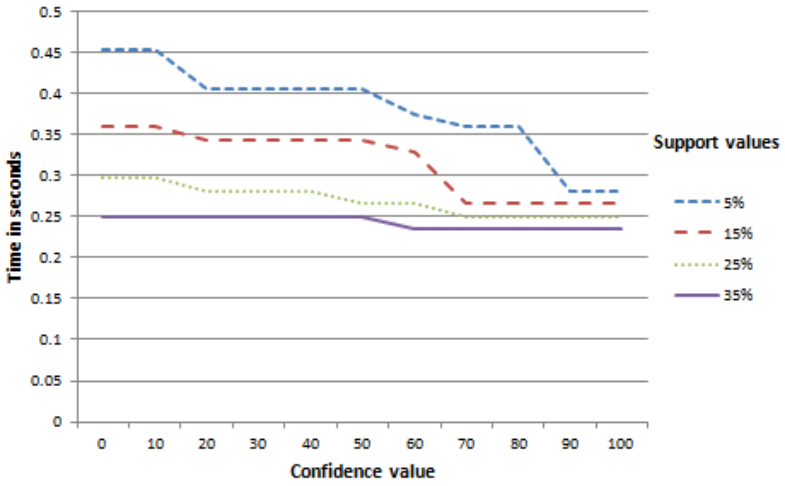


Figure 6.20: Measuring the execution time while varying the support and the confidence values of the mined rules.

the x-axis. We considered different support values ranging from 5% to 35%. Above 35%, the reported results were the same. The measured time is drawn as a line for each support value and for each confidence value.

We can obviously see that the execution time was decreasing when the confidence value was increasing. In fact, the number of used rules decreased since rules having a confidence value less than the minimum one were eliminated. The same reasoning was true for the different support values. In both cases, fewer applied rules required less execution time.

In Figure 6.21, we show the effect of varying the number of photos in a collection. Different measures are shown for each collection evaluated based on different confidence values. As observed on the graph, the execution time needed to process all the photos increased rapidly but this is acceptable when having reasonable social network size and/or number of shared photos.

Test 11: Time Analysis for co-referent relationship discovery

The objective of this test was to measure the temporal aspect while using the blocking technique for profile matching. We evaluated the time analysis under two different scenarios. We measured the execution time with and without using our blocking technique for profile matching. It was obvious that the task of social relationship type discovery added up some more process on the overall algorithm. However, this amount of time was not significant on the total time of process, as shown in the results. In the first scenario, we varied the percentage of co-referent contacts by increasing their percentage. As a result, the total number of contacts

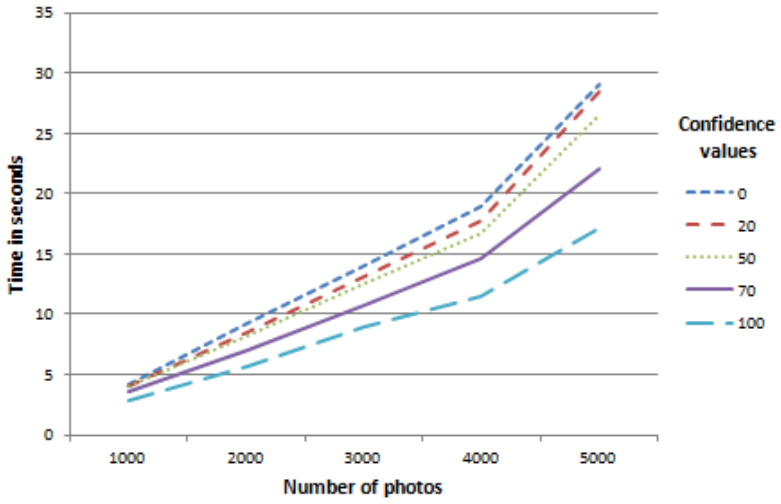


Figure 6.21: Measuring the execution time while varying the number of photos.

increased. In the second scenario, we varied the percentage of co-referent contacts by replacing non co-referent contacts with co-referent contacts. In the later scenario, the total number of contacts didn't change. The results of the first scenario is shown in Figure 6.22 and the results of the second scenario are shown in Figure 6.23.

As shown in the Figure 6.22 and Figure 6.23, the time needed for the overall process for the profile matching using the blocking technique was lower than the timing needed if all profiles are compared together without using blocks. This was true for almost all percentages except when the percentage of co-referent contacts was less than 3%. In fact, this is the worst case of our approach where all blocks were compared together, in addition to the time needed for relationship typed discovery.

To sum up, for social relationship discovery, we can observe a clear correlation between the number of rules to apply and the increase in the time needed to discover social relationship types. Meanwhile, the time execution is functionally proportional to the number of photos in a collection. As for the co-referent relationship discovery, we showed the utility of using the blocking optimization technique on time needed to compare and identify co-referent users.

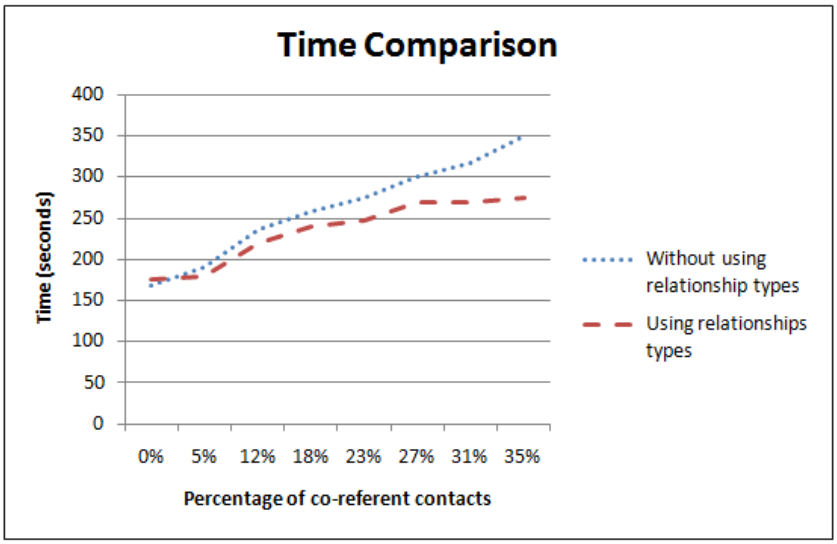


Figure 6.22: Time analysis with an increasing number of contacts.

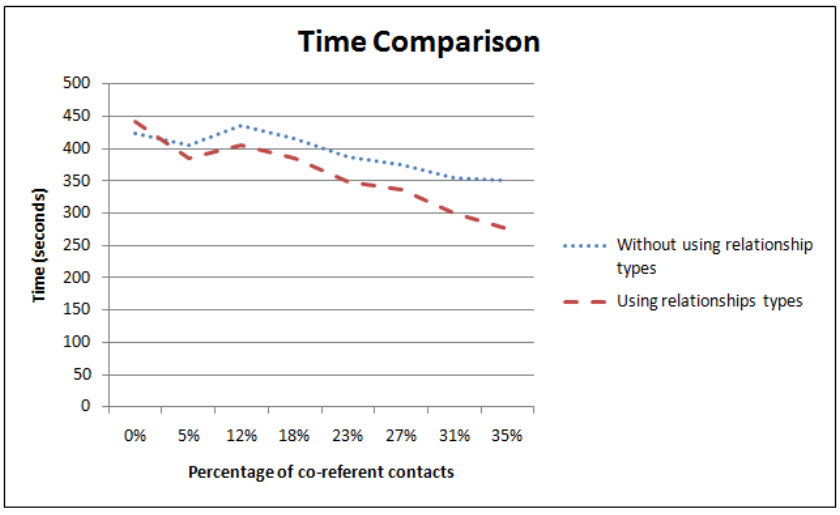


Figure 6.23: Time analysis with a fixed number of contacts.

6.3 Summary

In this chapter, we have presented our prototype designed to validate and demonstrate the practicability of our relationship discovery approach. We have described the set of real-world and synthetic datasets used in our evaluations. In addition, we have presented the results of the set of experiments conducted to validate the said approach. The aim of these tests was to experimentally verify the benefits and evaluate the usefulness in terms of validity, correctness, completeness, time execution, and relevance of our proposed relationship discovery algorithm along with the different types of used rules. The obtained results confirmed the relevance of our set of basic and derived rules, highlighting that the use of profiles and photos' information offers a significant benefit to the relationship discovery algorithm.

Chapter 7

Conclusion

"A little knowledge that acts is worth infinitely more than much knowledge that is idle." - Gibran Khalil Gibran

Enriching links with corresponding relationship types plays an important role in discovering the types of relationships between users. In fact, relationship type discovery emerges from the use of social network sites for a variety of purposes (e.g., to enhance privacy, facilitate relationship management, aggregate and enrich data, etc.). Yet, it is still a great challenge to provide appropriate means to identify automatically relationship types among social network users. In this work, we addressed relationship discovery 1) within the scope of the same social network to identify the social relationship types between a main user and her contacts, and 2) across different social network sites to identify co-referent users that refer to the same real-world contacts of a same main user. We proposed a rule-based approach to generate relevant rules to identify relationships.

The remainder of this chapter is organized as follows. Section 7.1 briefly reviews the main contributions of our work. In Section 7.2, we ping down several of our future research directions.

7.1 Contributions

The main contributions of this work consist of generating a set of basic rules and extending them using our proposed relationship type discovery approach. The contributions of this work are therefore summarized in the following:

1. We proposed a relationship discovery approach able to semantically enrich links with appropriate types between users. We investigated two link mining tasks: link type prediction and entity resolution. These two tasks are relatively newcomers on social network sites and studying these two tasks together is in fact a novel and interesting topic.

2. We introduced two types of rules that operate at different levels of granularity:

   (a) Intra-social rules: aim at identifying the appropriate social relationship types between a main user and her contacts within a single social network site.

   (b) Inter-social rules: aim at identifying the same real-world contacts of a main user having a user profile on two distinct social network sites.

3. We proposed two types of rules to identify social and co-referent relationships which are: basic rules, and derived rules. Basic and derived rules are both used as intra-social and inter-social rules. We detailed how to obtain each set of rules.

4. We devised two methodologies which are able to generate basic rules:

(a) Social relationship type methodology: to the best of our knowledge, this is the first methodology that is based on a crowdsourcing technique. It emphasizes the use of information available on users' profiles, in particular photos and their embedded metadata and related attributes. It is the first approach that explores photo features (metadata and attributes). In addition, we manually created a set of common sense based rules to enrich the set of basic rules.

(b) Co-referent relationship type methodology: to the best of our knowledge, this is the first methodology that takes into account all the attributes and features with dedicated distances given pair of social networks. This methodology builds up on the use of IFP to create the basic rules that incorporate the use of all profile attributes (including photos) and assign them with corresponding weights.

5. We proposed a method to create the so-called derived rules that are widely useful to remedy the problem of rule incompleteness. In fact, the set of derived rules can extend the set of basic rules and take into consideration the context of users, namely by identifying frequently used basic rules for each user.

6. We developed a prototype to validate and demonstrate the efficiency and reliability of our relationship discovery approach presented in this work. We also tested the relevance of our proposal using both real-world and synthetic datasets. In addition, we prepared a questionnaire to validate the correctness of our social relationships rules. We evaluated our approach using qualitative and quantitative measures.

Our publications:

International Journal:

E. Raad, R. Chbeir, and A. Dipanda, Discovering relationship types between users using profiles and shared photos on a social network, *Multimedia Tools and Applications*, pp. 1-30, 2011.

International Conference:

E. Raad, R. Chbeir, and A. Dipanda, User profile matching in social networks, 13th International Conference on Network-Based Information Systems (NBiS), pp. 297-304, 2010.

Book Chapter:

E. Raad, B. A. Bouna, and R. Chbeir, Bridging sensing and decision making in ambient intelligence environments, *Multimedia Techniques for Device and Ambient Intelligence*, J. Jeong and E. Damiani, Eds. Springer US, 2009, pp. 135-164.

7.2 Future Works

Analyzing social interactions among users is a complex problem, which becomes increasingly important with the rise of social networks. Although in this work we have addressed various aspects related to the relationship discovery, several other issues are yet to be addressed in the future. In the following, we present some of the possible research directions.

7.2.1 Improving Existing Approach

Detect Events

The social relationship discovery proposed in this work is capable to identify the type of relationships between a main user and her contacts using a set of rules. Inevitably, some relationship types can remain unlabeled due to missing values or lack of appropriate rules to identify these relationships. Given that social network users usually participate to common events and publish corresponding photos online, social relationship discovery approaches can build up on this to propose the adequate relationship type. In fact, event detection could be one of the promising solutions to improve relationship discovery and would be used as a recommender system for social network users.

Detect and Compare Clothing

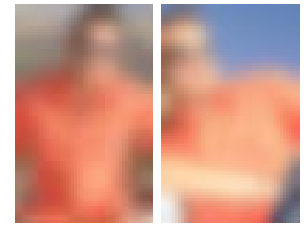
Modeling events in terms of date and location is important. However it is not always possible. In fact, event detection approaches based only on date and location information cannot cope with:

- missing date and location information
- manual setting of an event's duration
- events that take place in different locations (e.g., a wedding)

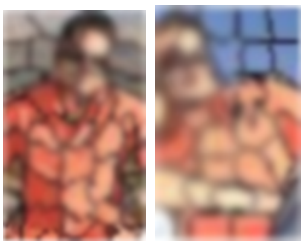
Incorporating additional cues to detect events is of considerable interest. One valuable cue for detecting events is the clothing of persons depicted in photos which could potentially enhance the detection of events and consequently be used within our social relationship discovery approach. In practice, persons usually do not change their clothing when they appear in a set of photos taken during a short period of time and then published online.

Clothing detection has been addressed in previous works [222] [223] [224]. Currently, we are in the process of investigating how to detect the clothing region within photos. In practice,

given two photos of the same person, the clothing detection algorithm processes as follows. First, each person's face is detected using a face detection algorithm [192]. Secondly, the body is cropped using heuristic measures based on empirical observations as described in [222]. Then, the cropped body is segmented into superpixels (small regions) using the normalized cuts approach [225]. Finally, each superpixel on the first photo is compared to its corresponding superpixel on the second photo based on their locations. Figure 7.1 shows some preliminary results that we obtained.



(a) Photo 1 - Cropped body (b) Photo 2 - Cropped body



(c) Photo 1 - with superpixels (d) Photo 2 - with superpixels

(e) Photo 1 - Detected clothing (f) Photo 2 - Detected clothing

Figure 7.1: Screenshot of a person depicted in two different photos (a) and (b), the results of applying the superpixel segmentation on both photos (c) and (d), and the detected clothing regions (e) and (f).

These results need further improvements in terms of quality and reliability. In our view, clothing detection represents an excellent initial step toward relevant event detection. We think

that our event detection will certainly enhance the results of our social relationship approach.

Incorporate a Visual Distance

Visual distance within photos can be seen as an evidence of the degree of intimacy between depicted persons. Studying the meaning of visual distance and what social meaning it underlines is an interesting topic for relationship type discovery. For example, best friends and family members tend to be physically closer to each other in photos. Visual distance must take into account different contextual parameters to include a wider number of the technical information available in Exif (such as flash, exposure time, subject distance, etc.). We are currently working on proposing a social closeness measure based on the visual distance between persons depicted in photos.

7.2.2 Improving Validation

Generate Common Sense Co-referent Rules

In this work, we emphasized on the use of rules to identify appropriate relationship types among social network users. However, in order to leverage the semantic web to extract more reliable rules based on the wisdom of a large community of users, one needs to further analyze social network interactions and trends. One area of future work will be to generate common sense rules for identifying co-referent users based on a crowdsourcing methodology. Research into solving this problem is already underway.

Ensure Rule Correctness

Giving to the main user the possibility to devise her own relationship discovery rules is not an easy task. It is therefore interesting to have an intelligent system that assists the main user so to avoid generating conflicting rules, identify unused rules and notify the end user, compute in real-time the minimum number of rules that the user must provide, etc. Future works should benefit greatly from the semantic web while using semantic-based reasoners to automatically infer appropriate rules and detect unwanted ones.

Prototype and Experiments

In addition, we will evaluate our approach by conducting further experimental investigations to estimate its efficiency on different relationship types (in addition to colleagues, relatives, and friends). We plan to provide methodologies to generate social basic rules according to different users' profiles. Actually, in this work we focused on generating social basic rules dedicated to adult social network users. In the future, we will study and evaluate the generated social

basic rules based on other factors: age (e.g., teenagers, seniors, etc.), region (e.g., country, GPS location, etc.), profession (e.g., nurses, engineers, etc.) We also plan to enhance our prototype so to automate different image processing techniques such as clothing detection, age and gender estimation, etc. In addition, we plan to provide a public web-based version of our prototype since it could help us to test and validate our obtained results.

7.2.3 Potential Application

Extending the Star Social Network

As it is described in this work, our social network is modeled as a star network. Extending this star social network to take into account links between contacts is an interesting aspect to study. Mutual contacts (friends, relatives, colleagues, etc.), relationship reciprocity, relationship transitivity, are all significant evidence that can help achieve better relationship discovery. We are also interested in identifying the most important factors related to users' behaviors to predict the type of social closeness. Exploiting networks that go beyond the star social network can help to better represent interactions with all mutual connections.

Enriching the Star Social Network

Social network sites are one source of information that can be used to identify relationship types. However, other rich sources exist and exploiting them can be of great boost to relationship type discovery. For instance, information available publicly on search engines are an interesting source of information. However, several elements must be analyzed when using search engines such as using the most relevant keywords when searching for a specific information, assigning appropriate weights to different criteria (e.g., the rank of the returned results, the source web site, etc.), combining different types of information (e.g., textual, photos, videos, etc.), etc. Another interesting aspect to consider in the future is to go beyond the use of photos for relationship type discovery. In fact, exploiting other multimedia sources (such as videos) can be interestingly useful for detecting relationship types.

Enforcing Users' Privacy

Enforcing privacy is an important problem to be resolved within future studies. One particular privacy threat is raised by the increasingly growing number of multimedia objects uploaded on social networks, in particular photos: 3 billion photos are uploaded each month on Facebook [1]. In fact, photos can be used to identify persons, as well as to infer additional information related to private life, friends, work information, habits, etc.

[1] http://www.litmanlive.co.uk/blog/2011/04/2010-by-the-numbers-stats-aplenty/

For instance, personal photos published on social networks may be used for inappropriate purposes: by employers to justify a decision to fire an employee, to check the backgrounds of potential employees, etc. Similarly, such available information, coupled with other profile content, may be used by family members, friends, or colleagues to check information related to sexuality, relationship status, or details of personal problems that owners might consider embarrassing if widely known [226]. Social network users are, however, often not aware of the audience that has access to their published information. Consequently, they are exposed to a number of privacy threats and risks [227].

Currently, social network users are facing different kinds of misuse cases regarding their privacy due to the lack of efficient access control models. Building up on the encouraging results of our relationship discovery approach, we are interested in extending this work to a relationship-based access control model to help solving this ever increasing privacy issue. We believe that such model will let social network users control the access to their personal information by only granting access to some contacts (e.g., friends) while denying the access of other contacts (e.g., colleagues).

References

[1] Nielsen, "State of the media: The social media report," http://www.nielsen.com/content/dam/corporate/us/en/reports-downloads/2011-Reports/nielsen-social-media-report.pdf, 2011, [Online; accessed 20-September-2011].

[2] J. Cranshaw, E. Toch, J. Hong, A. Kittur, and N. Sadeh, "Bridging the gap between physical location and online social networks," in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, ser. Ubicomp '10. New York, NY, USA: ACM, 2010, pp. 119–128.

[3] M. Roth, A. Ben-David, D. Deutscher, G. Flysher, I. Horn, A. Leichtberg, N. Leiser, Y. Matias, and R. Merom, "Suggesting friends using the implicit social graph," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 233–242.

[4] L. Tang and H. Liu, "Scalable learning of collective behavior based on sparse social dimensions," in *Proceeding of the 18th ACM conference on Information and knowledge management*, ser. CIKM '09. New York, NY, USA: ACM, 2009, pp. 1107–1116.

[5] U. Bojars, B. Heitmann, and E. Oren, "A prototype to explore content and context on social community sites," in *The Social Semantic Web 2007, Proceedings of the 1st Conference on Social Semantic Web (CSSW)*, ser. LNI, vol. 113. GI, 2007, pp. 47–58.

[6] M. J. Brzozowski, T. Hogg, and G. Szabo, "Friends and foes: ideological social networking," in *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ser. CHI '08. New York, NY, USA: ACM, 2008, pp. 817–820.

[7] T. Hogg, D. Wilkinson, G. Szabo, and M. Brzozowski, "Multiple relationship types in online communities and social networks," in *Proceedings of the AAAI Symposium on Social Information Processing*. AAAI, 2008, pp. 30–35.

[8] F. K. Ozenc and S. D. Farnham, “Life "modes" in social media,” in *Proceedings of the 2011 annual conference on Human factors in computing systems*, ser. CHI ’11. New York, NY, USA: ACM, 2011, pp. 561–570.

[9] R. Sarvas, D. M. Frohlich, R. Sarvas, and D. M. Frohlich, “The future of domestic photography,” in *From Snapshots to Social Media - The Changing Picture of Domestic Photography*, ser. Computer Supported Cooperative Work. Springer London, 2011, pp. 139–177.

[10] D. Easley and J. Kleinberg, *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.

[11] D. M. Boyd, “Friendster and publicly articulated social networking,” in *CHI ’04 extended abstracts on Human factors in computing systems*, ser. CHI EA ’04. ACM, 2004, pp. 1279–1282.

[12] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules in large databases,” in *Proceedings of the 20th International Conference on Very Large Data Bases*, ser. VLDB ’94. Morgan Kaufmann Publishers Inc., 1994, pp. 487–499.

[13] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[14] J. P. Scott, *Social Network Analysis: A Handbook*. SAGE Publications, Jan. 2000.

[15] L. Getoor and C. P. Diehl, “Link mining: a survey,” *ACM SIGKDD Explorations Newsletter*, vol. 7, pp. 3–12, December 2005.

[16] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.

[17] Santo and Fortunato, “Community detection in graphs,” *Physics Reports*, vol. 486, no. 3-5, pp. 75 – 174, 2010.

[18] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A. L. Barabási, “The large-scale organization of metabolic networks,” *Nature*, vol. 407, no. 6804, pp. 651–4, 2000.

[19] S. Maslov and K. Sneppen, “Specificity and stability in topology of protein networks,” *Science*, vol. 296, no. 5569, pp. 910–913, 2002.

[20] *The structure of scientific collaboration networks*, vol. 98, no. 2, 2001.

[21] A. Barabási, H. Jeong, R. Ravasz, Z. Néda, T. Vicsek, and A. Schubert, “On the topology of the scientific collaboration networks,” *Physica A*, vol. 311, pp. 590–614, 2002.

[22] A. Barrat, M. Barthélemy, R. Pastor-Satorras, and A. Vespignani, "The architecture of complex weighted networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 11, pp. 3747–3752, 2004.

[23] X. Jiang, H. Xiong, C. Wang, and A.-H. Tan, "Mining globally distributed frequent subgraphs in a single labeled graph," *Data Knowl. Eng.*, vol. 68, pp. 1034–1058, October 2009.

[24] Y. Sun, Y. Yu, and J. Han, "Ranking-based clustering of heterogeneous information networks with star network schema," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '09. New York, NY, USA: ACM, 2009, pp. 797–806.

[25] M. San Martín and C. Gutierrez, "Representing, querying and transforming social networks with rdf/sparql," in *Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications*, ser. ESWC 2009 Heraklion. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 293–307.

[26] A. M. Kaplan and M. Haenlein, "Users of the world, unite! the challenges and opportunities of social media," *Business Horizons*, vol. 53, no. 1, pp. 59–68, January 2010.

[27] J. R. Tyler, D. M. Wilkinson, and B. A. Huberman, *Email as spectroscopy: automated discovery of community structure within organizations.* Deventer, The Netherlands, The Netherlands: Kluwer, B.V., 2003, pp. 81–96.

[28] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Networks*, vol. 25, no. 3, pp. 211–230, 2003.

[29] T. Miki, S. Nomura, and T. Ishida, "Semantic web link analysis to discover social relationships in academic communities," in *Proceedings of the The 2005 Symposium on Applications and the Internet.* Washington, DC, USA: IEEE Computer Society, 2005, pp. 38–45.

[30] B. H. Junker and F. Schreiber, *Analysis of Biological Networks.* Wiley-Interscience, 2008.

[31] B. Schneier, "A taxonomy of social networking data," *Security Privacy, IEEE*, vol. 8, no. 4, p. 88, 2010.

[32] J. Diesner, T. Frantz, and K. Carley, "Communication networks from the enron email corpus "it's always about the people. enron is no different"," *Computational & Mathematical Organization Theory*, vol. 11, pp. 201–228, 2005.

[33] A. A. Nanavati, S. Gurumurthy, G. Das, D. Chakraborty, K. Dasgupta, S. Mukherjea, and A. Joshi, "On the structural properties of massive telecom call graphs: findings and implications," in *Proceedings of the 15th ACM international conference on Information and knowledge management*, ser. CIKM '06. New York, NY, USA: ACM, 2006, pp. 435–444.

[34] J. Tang, R. Jin, and J. Zhang, "A topic modeling approach and its integration into the random walk framework for academic search," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 1055–1060.

[35] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, ser. IMC '07. New York, NY, USA: ACM, 2007, pp. 29–42.

[36] N. Memon, D. L. Hicks, H. L. Larsen, and M. A. Uqaili, "Understanding the structure of terrorist networks," *International Journal of Business Intelligence and Data Mining*, vol. 2, no. 4, pp. 401–425, 2007.

[37] R. M. Christley, G. L. Pinchbeck, R. G. Bowers, D. Clancy, N. P. French, R. Bennett, and J. Turner, "Infection in social networks: Using network analysis to identify high-risk individuals," *American Journal of Epidemiology*, vol. 162, no. 10, pp. 1024–1031, 15 November 2005.

[38] D. Koschützki, K. Lehmann, L. Peeters, S. Richter, D. Tenfelde-Podehl, and O. Zlotowski, "Centrality indices," in *Network Analysis*, ser. Lecture Notes in Computer Science, U. Brandes and T. Erlebach, Eds. Springer Berlin / Heidelberg, 2005, vol. 3418, pp. 16–61.

[39] L. C. Freeman, "Centrality in social networks: Conceptual clarification," *Social Networks*, vol. 1, pp. 215–239, 1979.

[40] P. Bonacich, "Power and centrality: A family of measures," *American Journal of Sociology*, vol. 92, no. 5, pp. 1170–1182, 1987.

[41] U. Brandes, P. Kenis, and D. Wagner, "Communicating centrality in policy network drawings," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 9, no. 2, pp. 241–253, april-june 2003.

[42] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '03. New York, NY, USA: ACM, 2003, pp. 137–146.

[43] N. Memon, H. L. Larsen, D. L. Hicks, and N. Harkiolakis, "Detecting hidden hierarchy in terrorist networks: Some case studies," in *Proceedings of the IEEE ISI 2008 PAISI, PACCF, and SOCO international workshops on Intelligence and Security Informatics*, ser. PAISI, PACCF and SOCO '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 477–489.

[44] A. Chin and M. Chignell, "Identifying communities in blogs: roles for social network analysis and survey instruments," *International Journal of Web Based Communities*, vol. 3, no. 3, pp. 345–363, 2007.

[45] S. Wuchty and P. F. Stadler, "Centers of complex networks," *Journal of theoretical biology*, vol. 223, no. 1, pp. 45–53, 2003.

[46] A. Bavelas, "A mathematical model of group structure," *Human Organizations*, vol. 7, pp. 16–30, 1948.

[47] B. Singh and N. Gupte, "Congestion and decongestion in a communication network," *Physical Review E*, vol. 71, no. 5, p. 055103, 2005.

[48] M. Ramirez Ortiz, J. Caballero Hoyos, and M. Ramirez Lopez, "The social networks of academic performance in a student context of poverty in mexico," *Social networks*, vol. 26, no. 2, pp. 175–188, 2004.

[49] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Technical Report 1999-66, November 1999.

[50] M. Shaikh, J. Wang, Z. Yang, and Y. Song, "Graph structural mining in terrorist networks," in *Advanced Data Mining and Applications*, ser. Lecture Notes in Computer Science, R. Alhajj, H. Gao, X. Li, J. Li, and O. Zaïane, Eds. Springer Berlin / Heidelberg, 2007, vol. 4632, pp. 570–577.

[51] G. Erkan and D. R. Radev, "Lexrank: graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research (JAIR)*, vol. 22, pp. 457–479, December 2004.

[52] B. Ball, B. Karrer, and M. E. J. Newman, "Efficient and principled method for detecting communities in networks," *Physical Review E*, vol. 84, p. 036103, Sep 2011.

[53] D. Jensen and H. Goldberg, *AAAI Fall Symposium on AI and Link Analysis.* AAAI Press, 1998.

[54] S. Chakrabarti, *Mining the Web: Discovering Knowledge from Hypertext Data.* Morgan-Kauffman, 2002.

[55] S. Dzeroski, *Relational Data Mining*, 1st ed., N. Lavrac, Ed. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2001.

[56] D. J. Cook and L. B. Holder, "Graph-based data mining," *IEEE Intelligent Systems*, vol. 15, pp. 32–41, March 2000.

[57] J. Apostolakis, "An introduction to data mining," in *Data Mining in Crystallography*, ser. Structure & Bonding, D. W. M. Hofmann and L. N. Kuleshova, Eds. Springer Berlin Heidelberg, 2010, vol. 134, pp. 1–35.

[58] X. Liu, J. Bollen, M. L. Nelson, and H. Van de Sompel, "Co-authorship networks in the digital library research community," *Inf. Process. Manage.*, vol. 41, pp. 1462–1480, December 2005.

[59] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, pp. 107–117, April 1998.

[60] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," in *Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms*, ser. SODA '98. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1998, pp. 668–677.

[61] P. Li, Z. Li, H. Liu, J. He, and X. Du, "Using link-based content analysis to measure document similarity effectively," in *Advances in Data and Web Management*, ser. Lecture Notes in Computer Science, Q. Li, L. Feng, J. Pei, S. Wang, X. Zhou, and Q.-M. Zhu, Eds. Springer Berlin / Heidelberg, 2009, vol. 5446, pp. 455–467.

[62] T. Carpenter, G. Karakostas, and D. Shallcross, "Practical issues and algorithms for analyzing terrorist networks," *Proceedings of the Western Simulation MultiConference*, 2002.

[63] J. Scripps, R. Nussbaum, P.-N. Tan, and A.-H. Esfahanian, "Link-based network mining," in *Structural Analysis of Complex Networks*, M. Dehmer, Ed. Birkhäuser Boston, 2011, pp. 403–419.

[64] J. Karamon, Y. Matsuo, H. Yamamoto, and M. Ishizuka, "Generating social network features for link-based classification," in *Proceedings of the 11th European conference on*

Principles and Practice of Knowledge Discovery in Databases, ser. PKDD 2007. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 127–139.

[65] E. Segal, H. Wang, and D. Koller, "Discovering molecular pathways from protein interaction and gene expression data," *Bioinformatics*, vol. 19, no. suppl 1, pp. i264–i272, 2003.

[66] E. Stattner and N. Vidot, "Social network analysis in epidemiology: Current trends and perspectives," in *Research Challenges in Information Science (RCIS), 2011 Fifth International Conference on*, may 2011, pp. 1 –11.

[67] S. Chakrabarti, B. Dom, and P. Indyk, "Enhanced hypertext categorization using hyperlinks," in *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, ser. SIGMOD '98. New York, NY, USA: ACM, 1998, pp. 307–318.

[68] I. Bhattacharya and L. Getoor, "Deduplication and group detection using links," in *ACM SIGKDD Workshop on Link Analysis and Group Detection (LinkKDD)*, 2004.

[69] A. Fard and M. Ester, "Collaborative mining in multiple social networks data for criminal group discovery," in *Computational Science and Engineering, 2009. CSE '09. International Conference on*, vol. 4, aug. 2009, pp. 582 –587.

[70] G. Barbier and H. Liu, "Data mining in social media," in *Social Network Data Analytics*, C. C. Aggarwal, Ed. Springer US, 2011, pp. 327–352.

[71] I. Fellegi and A. B. Sunter, "A theory for record linkage," *Journal of the American Statistical Association*, vol. 64, no. 328, pp. 1183–1210, 1969.

[72] A. Monge and C. Elkan, "An efficient domain-independent algorithm for detecting approximately duplicate database records," in *SIGMOD workshop on data mining and knowledge discovery*, 1997, pp. 23–29.

[73] M. A. Hernández and S. J. Stolfo, "Real-world data is dirty: Data cleansing and the merge/purge problem," *Data Mining and Knowledge Discovery*, vol. 2, pp. 9–37, 1998, 10.1023/A:1009761603038.

[74] X. Dong, A. Halevy, and J. Madhavan, "Reference reconciliation in complex information spaces," in *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, ser. SIGMOD '05. ACM, 2005, pp. 85–96.

[75] S. Tejada, C. A. Knoblock, and S. Minton, "Learning object identification rules for information integration," *Information Systems*, vol. 26, pp. 607–633, 2001.

[76] L. Stein, "Integrating biological databases," *Nature Reviews Genetics*, vol. 4, no. 5, pp. 337–345, 2003.

[77] I. Bhattacharya and L. Getoor, "Iterative record linkage for cleaning and integration," in *Proceedings of the 9th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, ser. DMKD '04. ACM, 2004, pp. 11–18.

[78] S. P. Ponzetto and M. Strube, "Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, ser. HLT-NAACL '06. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006, pp. 192–199.

[79] E. Raad, R. Chbeir, and A. Dipanda, "User profile matching in social networks," *Network-Based Information Systems (NBiS), 2010 13th International Conference on*, pp. 297–304, 2010.

[80] J. Sleeman and T. Finin, "A machine learning approach to linking foaf instances," in *Proceedings of the AAAI Spring Symposium on Linked Data Meets Artificial Intelligence*. AAAI Press, 2010.

[81] B. Taskar, M. Wong, P. Abbeel, and D. Koller, "Link prediction in relational data," in *Advances in Neural Information Processing Systems (NIPS) 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2003.

[82] J. OMadadhain, J. Hutchins, and P. Smyth, "Prediction and ranking algorithms for event-based network data," *ACM SIGKDD Explorations Newsletter*, vol. 7, pp. 23–30, 2005.

[83] H. Yu, A. Paccanaro, V. Trifonov, and M. Gerstein, "Predicting interactions in protein networks by completing defective cliques," *Bioinformatics*, vol. 22, no. 7, pp. 823–829, 2006.

[84] Z. Huang, X. Li, and H. Chen, "Link prediction approach to collaborative filtering," in *Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries*, ser. JCDL '05. New York, NY, USA: ACM, 2005, pp. 141–142.

[85] E. Zheleva, L. Getoor, J. Golbeck, and U. Kuter, "Using friendship ties and family circles for link prediction," in *Proceedings of the Second international conference on Advances in social network mining and analysis*, ser. SNAKDD'08. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 97–113.

[86] G. M. Namata, H. Sharara, and L. Getoor, "A survey of link mining tasks for analyzing noisy and incomplete networks," in *Link Mining: Models, Algorithms, and Applications*, P. S. S. Yu, J. Han, and C. Faloutsos, Eds. Springer New York, 2010, pp. 107–133.

[87] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[88] P. Chaiwanarom and C. Lursinsap, "Link completion using prediction by partial matching," in *Communications and Information Technologies, 2008. ISCIT 2008. International Symposium on*, oct. 2008, pp. 675 –680.

[89] V. Carvalho and W. Cohen, "Preventing information leaks in email," in *Proceedings of the Seventh SIAM International Conference on Data Mining (SDM)*, 2007.

[90] Z. Huang and D. Zeng, "A link prediction approach to anomalous email detection," in *Systems, Man and Cybernetics, 2006. SMC '06. IEEE International Conference on*, vol. 2, oct. 2006, pp. 1131 –1136.

[91] L. Getoor, "Link mining: a new data mining challenge," *SIGKDD Explorations Newsletter*, vol. 5, pp. 84–89, July 2003.

[92] T. Zhang, H. Chao, C. Willis, and D. Tretter, "Consumer image retrieval by estimating relation tree from family photo collections," in *Proceedings of the ACM International Conference on Image and Video Retrieval*, ser. CIVR '10. New York, NY, USA: ACM, 2010, pp. 143–150.

[93] E. Raad, R. Chbeir, and A. Dipanda, "Discovering relationship types between users using profiles and shared photos in a social network," *Multimedia Tools and Applications*, pp. 1–30, 2011.

[94] X. Yan, X. J. Zhou, and J. Han, "Mining closed relational graphs with connectivity constraints," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, ser. KDD '05. New York, NY, USA: ACM, 2005, pp. 324–333.

[95] G. Ciriello and C. Guerra, "A review on models and algorithms for motif discovery in proteinŰprotein interaction networks," *Briefings in Functional Genomics & Proteomics*, vol. 7, no. 2, pp. 147–156, 2008.

[96] H. Cheng, D. Lo, Y. Zhou, X. Wang, and X. Yan, "Identifying bug signatures using discriminative graph mining," in *Proceedings of the eighteenth international symposium*

on Software testing and analysis, ser. ISSTA '09. New York, NY, USA: ACM, 2009, pp. 141–152.

[97] M. Last, A. Markov, and A. Kandel, "Multi-lingual detection of terrorist content on the web," in *Intelligence and Security Informatics*, ser. Lecture Notes in Computer Science, H. Chen, F.-Y. Wang, C. Yang, D. Zeng, M. Chau, and K. Chang, Eds. Springer Berlin / Heidelberg, 2006, vol. 3917, pp. 16–30.

[98] V. E. Lee, N. Ruan, R. Jin, and C. Aggarwal, "A survey of algorithms for dense subgraph discovery," in *Managing and Mining Graph Data*, ser. The Kluwer International Series on Advances in Database Systems, C. C. Aggarwal, H. Wang, and A. K. Elmagarmid, Eds. Springer US, 2010, vol. 40, pp. 303–336.

[99] K. M. Borgwardt, C. S. Ong, S. Schönauer, S. V. N. Vishwanathan, A. J. Smola, and H.-P. Kriegel, "Protein function prediction via graph kernels," *Bioinformatics*, vol. 21, no. suppl 1, pp. i47–i56, 2005.

[100] M. desJardins and M. E. Gaston, "Speaking of relations: Connecting statistical relational learning and multi-agent systems," 2006.

[101] C. Yu and J. Yu, "Mining behavior graphs for" backtrace" of noncrashing bugs," in *Proceedings of the Fifth SIAM International Conference on Data Mining*, vol. 119. Society for Industrial Mathematics, 2005, p. 286.

[102] M. Deshpande, M. Kuramochi, and G. Karypis, "Frequent sub-structure-based approaches for classifying chemical compounds," in *Proceedings of the Third IEEE International Conference on Data Mining*, ser. ICDM '03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 35–42.

[103] R. D. King, S. H. Muggleton, A. Srinivasan, and M. J. Sternberg, "Structure-activity relationships derived by machine learning: the use of atoms and their bond connectivities to predict mutagenicity by inductive logic programming," *Proceedings of the National Academy of Sciences*, vol. 93, no. 1, pp. 438–442, 1996.

[104] T. Gärtner, "Exponential and geometric kernels for graphs," in *NIPS Workshop on Unreal Data: Principles of Modeling Nonvectorial Data*, 2002.

[105] H. Kashima and A. Inokuchi, "Kernels for graph classification," *ICDM Workshop on Active Mining*, 2002.

[106] K. Tsuda and H. Saigo, "Graph classification," in *Managing and Mining Graph Data*, 2010, pp. 337–363.

[107] C. Nguyen and H. Mamitsuka, "Kernels for link prediction with latent feature models," in *Machine Learning and Knowledge Discovery in Databases*, ser. Lecture Notes in Computer Science, D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis, Eds. Springer Berlin / Heidelberg, 2011, vol. 6912, pp. 517–532.

[108] D. Zhou, E. Manavoglu, J. Li, C. L. Giles, and H. Zha, "Probabilistic models for discovering e-communities," in *Proceedings of the 15th international conference on World Wide Web*, ser. WWW '06. New York, NY, USA: ACM, 2006, pp. 173–182.

[109] A. B. N. Pathak and K. Erickson, "Social topic models for community extraction," in *The 2nd SNA-KDD Workshop Š08 (SNA-KDDŠ08)*, Las Vegas, Nevada, USA, 2008.

[110] M. S. Martin and C. Gutierrez, "Personal management of social networks data," in *Proceedings of the 2009 International Conference on Computational Science and Engineering - Volume 04*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 765–770.

[111] A. Lampinen, V. Lehtinen, A. Lehmuskallio, and S. Tamminen, "We're in it together: interpersonal management of disclosure in social network services," in *Proceedings of the 2011 annual conference on Human factors in computing systems*, ser. CHI '11. New York, NY, USA: ACM, 2011, pp. 3217–3226.

[112] T. Bhuiyan, A. Josang, and Y. Xu, "Trust and reputation management in web-based social network," *Web Intelligence and Intelligent Agents*, pp. 207–232, 2010.

[113] Z. Stone, T. Zickler, and T. Darrell, "Autotagging facebook: Social network context improves photo annotation," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, june 2008, pp. 1 –8.

[114] J. Hartline, V. Mirrokni, and M. Sundararajan, "Optimal marketing strategies over social networks," in *Proceeding of the 17th international conference on World Wide Web*, ser. WWW '08. New York, NY, USA: ACM, 2008, pp. 189–198.

[115] B. Carminati and E. Ferrari, "Privacy-aware access control in social networks: Issues and solutions," in *Privacy and Anonymity in Information Management Systems*, ser. Advanced Information and Knowledge Processing, J. Nin, J. Herranz, L. Jain, and X. Wu, Eds. Springer London, 2010, vol. 0, pp. 181–195.

[116] A. C. Squicciarini, M. Shehab, and J. Wede, "Privacy policies for shared content in social network sites," *The VLDB Journal*, vol. 19, pp. 777–796, December 2010.

[117] P. Wu and F. Tang, "Improving face clustering using social context," in *Proceedings of the international conference on Multimedia*, ser. MM '10. New York, NY, USA: ACM, 2010, pp. 907–910.

[118] M. Hernández and S. Stolfo, "Real-world data is dirty: Data cleansing and the merge/purge problem," *Data mining and knowledge discovery*, vol. 2, no. 1, pp. 9–37, 1998.

[119] V. Sehgal, L. Getoor, and P. D. Viechnicki, "Entity resolution in geospatial data integration," in *Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*, ser. GIS '06. New York, NY, USA: ACM, 2006, pp. 83–90.

[120] J. Bleiholder and F. Naumann, "Data fusion," *ACM Comput. Surv.*, vol. 41, pp. 1:1–1:41, January 2009.

[121] J. Tang, L. Yao, D. Zhang, and J. Zhang, "A combination approach to web user profiling," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 5, pp. 2:1–2:44, December 2010.

[122] H. Garcia-Molina, "Entity resolution: Overview and challenges," in *Conceptual Modeling Ű ER 2004*, ser. Lecture Notes in Computer Science, P. Atzeni, W. Chu, H. Lu, S. Zhou, and T.-W. Ling, Eds. Springer Berlin / Heidelberg, 2004, vol. 3288, pp. 1–2.

[123] H. B. Newcombe, J. M. Kennedy, S. J. Axfordd, and A. P. James, "Automatic linkage of vital records." *Science*, vol. 130, pp. 954–959, 1959.

[124] R. Ananthakrishna, S. Chaudhuri, and V. Ganti, "Eliminating fuzzy duplicates in data warehouses," in *Proceedings of the 28th international conference on Very Large Data Bases*, ser. VLDB '02. VLDB Endowment, 2002, pp. 586–597.

[125] D. V. Kalashnikov, S. Mehrotra, and Z. Chen, "Exploiting relationships for domain-independent data cleaning," in *SIAM International Conference on Data Mining (SIAM SDM)*, Newport Beach, CA, USA, April 21–23 2005.

[126] I. Bhattacharya and L. Getoor, *Entity Resolution in Graphs*. John Wiley & Sons, Inc., 2006, pp. 311–344.

[127] P. Singla and P. Domingos, "Entity resolution with markov logic," in *Proceedings of the Sixth International Conference on Data Mining*, ser. ICDM '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 572–582.

[128] D. Dey, S. Sarkar, and P. De, "A distance-based approach to entity reconciliation in heterogeneous databases," *IEEE Trans. on Knowl. and Data Eng.*, vol. 14, pp. 567–582, 2002.

[129] P. Ravikumar and W. W. Cohen, "A hierarchical graphical model for record linkage," in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, ser. UAI '04. AUAI Press, 2004, pp. 454–461.

[130] A. Culotta and A. McCallum, "Joint deduplication of multiple record types in relational data," in *Proceedings of the 14th ACM international conference on Information and knowledge management*, ser. CIKM '05. ACM, 2005, pp. 257–258.

[131] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[132] M. A. Jaro, "Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida," *Journal of the American Statistical Association*, vol. 84, no. 406, pp. 414–420, 1989.

[133] D. Gusfield, *Algorithms on strings, trees, and sequences: computer science and computational biology.* Cambridge University Press, 1997.

[134] I. Bhattacharya and L. Getoor, "Collective entity resolution in relational data," *ACM Trans. Knowl. Discov. Data*, vol. 1, March 2007.

[135] M. Richardson and P. Domingos, "Markov logic networks," *Machine Learning*, vol. 62, no. 1, pp. 107–136, 2006.

[136] S. E. Whang, D. Menestrina, G. Koutrika, M. Theobald, and H. Garcia-Molina, "Entity resolution with iterative blocking," in *SIGMOD 2009.* Stanford, June 2009.

[137] S. Bechhofer, F. Van Harmelen, J. Hendler, I. Horrocks, D. McGuinness, P. Patel-Schneider, L. Stein *et al.*, "Owl web ontology language reference," *W3C recommendation*, vol. 10, pp. 2006–01, 2004.

[138] L. Ding, L. Zhou, T. Finin, and A. Joshi, "How the semantic web is being used: An analysis of FOAF documents," in *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, 2005, pp. 113c–113c.

[139] P. Mika, "Flink: Semantic web technology for the extraction and analysis of social networks," *Journal of web semantics*, vol. 3, pp. 211—223, 2005.

[140] J. Golbeck and M. Rothstein, "Linking social networks on the web with foaf: a semantic web case study," in *Proceedings of the 23rd national conference on Artificial intelligence - Volume 2*. AAAI Press, 2008, pp. 1138–1143.

[141] S. Bortoli, H. Stoermer, P. Bouquet, and H. Wache, "Foaf-o-matic - solving the identity problem in the foaf network," in *In Proceedings of the Fourth Italian Semantic Web Workshop (SWAP2007)*, 2007.

[142] L. Shi, D. Berrueta, S. Fernandez, L. Polo, S. Fernandez, and A. Asturias, "Smushing rdf instances: are alice and bob the same open source developer?" in *ISWC2008 workshop on Personal Identification and Collaborations: Knowledge Mediation and Extraction (PICKME 2008)*, 2009.

[143] A. Hogan, "The expertfinder corpus 2007 for the benchmarking and development of expert-finding systems," in *First International ExpertFinder Workshop*, 2007.

[144] M. Rowe and F. Ciravegna, "Disambiguating identity through social circles and social data," in *Collective Intelligence Workshop ESWC 2008*, 2008.

[145] C. Zhou, H. Chen, and T. Yu, "Learning a probabilistic semantic model from heterogeneous social networks for relationship identification," in *Tools with Artificial Intelligence, 2008. ICTAI '08. 20th IEEE International Conference on*, vol. 1, nov. 2008, pp. 343 –350.

[146] P. Bouquet, H. Stoermer, and D. Giacomuzzi, "Okkam: Enabling a web of entities," in *Proceedings of the WWW2007 Workshop i3: Identity, Identifiers and Identification, Banff, Canada, May 8 2007*, ser. CEUR Workshop Proceedings, ISSN 1613-0073, P. Bouquet, H. Stoermer, G. Tummarello, and H. Halpin, Eds., 2007.

[147] C. L. Giles, K. D. Bollacker, and S. Lawrence, "Citeseer: an automatic citation indexing system," in *Proceedings of the third ACM conference on Digital libraries*, ser. DL '98. New York, NY, USA: ACM, 1998, pp. 89–98.

[148] M. Ley, "The dblp computer science bibliography: Evolution, research issues, perspectives," in *Proceedings of the 9th International Symposium on String Processing and Information Retrieval*, ser. SPIRE 2002. London, UK: Springer-Verlag, 2002, pp. 1–10.

[149] F. Saïs, N. Pernelle, and M.-C. Rousset, "L2R: A Logical Method for Reference Reconciliation," in *Twenty-Second AAAI Conference on Artificial Intelligence*, Vancouver, British Columbia, Canada, Jul. 2007, pp. 329–334.

[150] V. Vapnik, "Statistical learning theory," 1998.

[151] U. Bojārs, J. Breslin, V. Peristeras, G. Tummarello, and S. Decker, "Interlinking the social web with semantics," *IEEE Intelligent Systems*, pp. 29–40, 2008.

[152] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the twelfth international conference on Information and knowledge management*, ser. CIKM '03. New York, NY, USA: ACM, 2003, pp. 556–559.

[153] R. Parimi and D. Caragea, "Predicting friendship links in social networks using a topic modeling approach," in *Advances in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, J. Huang, L. Cao, and J. Srivastava, Eds. Springer Berlin / Heidelberg, 2011, vol. 6635, pp. 75–86.

[154] H. Kautz, B. Selman, and M. Shah, "The hidden web," *AI Magazine*, vol. 18, pp. 27–36, 1997.

[155] R. Rowe, G. Creamer, S. Hershkop, and S. J. Stolfo, "Automated social hierarchy detection through email network analysis," in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, ser. WebKDD/SNA-KDD '07. New York, NY, USA: ACM, 2007, pp. 109–117.

[156] M. A. Stefanone and G. Gay, "Structural reproduction of social networks in computer-mediated communication forums," *Behav. Inf. Technol.*, vol. 27, pp. 97–106, March 2008.

[157] M. Bilgic, G. M. Namata, and L. Getoor, "Combining collective classification and link prediction," in *Proceedings of the Seventh IEEE International Conference on Data Mining Workshops*, ser. ICDMW '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 381–386.

[158] Y. Jin, Y. Matsuo, and M. Ishizuka, "Extracting social networks among various entities on the web," in *Proceedings of the 4th European conference on The Semantic Web: Research and Applications*, ser. ESWC '07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 251–266.

[159] M.-S. Lin and H.-H. Chen, "Labeling categories and relationships in an evolving social network," in *Proceedings of the IR research, 30th European conference on Advances in*

information retrieval, ser. ECIR'08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 77–88.

[160] Y. Matsuo, J. Mori, M. Hamasaki, T. Nishimura, H. Takeda, K. Hasida, and M. Ishizuka, "Polyphonet: An advanced social network extraction system from the web," *Web Semant.*, vol. 5, pp. 262–278, December 2007.

[161] J. Mori, M. Ishizuka, and Y. Matsuo, "Extracting keyphrases to represent relations in social networks from web," in *Proceedings of the 20th international joint conference on Artifical intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007, pp. 2820–2825.

[162] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., 1986.

[163] H. Wang and L. Sun, "Trust-involved access control in collaborative open social networks," in *Proceedings of the 2010 Fourth International Conference on Network and System Security*, ser. NSS '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 239–246.

[164] W. Tang, H. Zhuang, and J. Tang, "Learning to infer social ties in large networks," in *Proceedings of the ECML/PKDD 2011*, 2011.

[165] S. Golder, "Measuring social networks with digital photograph collections," in *Proceedings of the nineteenth ACM conference on Hypertext and hypermedia*, ser. HT '08. New York, NY, USA: ACM, 2008, pp. 43–48.

[166] P. Wu and D. Tretter, "Close & closer: social cluster and closeness from photo collections," in *Proceedings of the 17th ACM international conference on Multimedia*, ser. MM '09. New York, NY, USA: ACM, 2009, pp. 709–712.

[167] P. Singla, H. Kautz, J. Luo, and A. Gallagher, "Discovery of social relationships in consumer photo collections using markov logic," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, june 2008, pp. 1 –7.

[168] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in *Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*, ser. HUC '99. London, UK: Springer-Verlag, 1999, pp. 304–307.

[169] P. Bouquet, H. Stoermer, and B. Bazzanella, "An entity name system (ENS) for the semantic web," in *The Semantic Web: Research and Applications*, 2008, pp. 258–272.

[170] A. Lanitis, C. Taylor, and T. Cootes, "Toward automatic simulation of aging effects on face images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 4, pp. 442–455, 2002.

[171] S. Baluja and H. Rowley, "Boosting sex identification performance," *International Journal of Computer Vision*, vol. 71, no. 1, pp. 111–119, 2007.

[172] T. Berners-Lee, D. Connolly, L. Kagal, Y. Scharf, and J. Hendler, "N3logic: A logical framework for the world wide web," *Theory and Practice of Logic Programming (TPLP)*, vol. 8, no. 3, pp. 249–269, 2008.

[173] R. W. Sinnott, "Virtues of the haversine," *Sky and Telescope*, vol. 68, p. 159, dec 1984.

[174] A. Dempster, "A generalization of the bayesian inference," *Journal of the Royal Statistical Society, Series B*, vol. 30, pp. 205–247, 1968.

[175] G. Shafer, *A mathematical theory of evidence*. Princeton university press, 1976, vol. 1.

[176] W. W. Cohen, P. D. Ravikumar, and S. E. Fienberg, "A comparison of string distance metrics for name-matching tasks," in *Proceedings of the IJCAI-2003 Workshop on Information Integration on the Web (IIWeb-03)*, 2003, pp. 73–78.

[177] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," in *Proceedings of the 20th international joint conference on Artifical intelligence*, ser. IJCAI'07. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007, pp. 1606–1611.

[178] T. Green and A. Quigley, "Perception of online social networks," in *Mining and Analyzing Social Networks*, ser. Studies in Computational Intelligence, I.-H. Ting, H.-J. Wu, and T.-H. Ho, Eds. Springer Berlin / Heidelberg, 2010, vol. 288, pp. 91–106.

[179] N. Li, N. Zhang, and S. Das, "Preserving relation privacy in online social network data," *Internet Computing, IEEE*, vol. 15, no. 3, pp. 35 –42, may-june 2011.

[180] A. Wu, J. M. DiMicco, and D. R. Millen, "Detecting professional versus personal closeness using an enterprise social network site," in *Proceedings of the 28th international conference on Human factors in computing systems*, ser. CHI '10. New York, NY, USA: ACM, 2010, pp. 1955–1964.

[181] L. Zhang, J. Ma, C. Cui, and P. Li, "Active learning through notes data in flickr: an effortless training data acquisition approach for object localization," in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ser. ICMR '11. New York, NY, USA: ACM, 2011, pp. 46:1–46:8. [Online]. Available: http://doi.acm.org/10.1145/1991996.1992042

[182] B. Collins, J. Deng, K. Li, and L. Fei-Fei, "Towards scalable dataset construction: An active learning approach," in *Computer Vision Ű ECCV 2008*, ser. Lecture Notes in Computer Science, D. Forsyth, P. Torr, and A. Zisserman, Eds. Springer Berlin / Heidelberg, 2008, vol. 5302, pp. 86–98.

[183] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, oct. 2007, pp. 1 –8.

[184] J. Fan, Y. Shen, N. Zhou, and Y. Gao, "Harvesting large-scale weakly-tagged image databases from the web," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 802–809.

[185] M. J. Huiskes and M. S. Lew, "The mir flickr retrieval evaluation," in *Multimedia Information Retrieval*, 2008, pp. 39–43.

[186] J. Howe, "The rise of crowdsourcing," *Wired magazine*, vol. 14, no. 6, pp. 1–4, 2006.

[187] J. Surowiecki, *The wisdom of crowds: Why the many are smarter than the few.* Doubleday Books, 2004.

[188] P. Welinder and P. Perona, "Online crowdsourcing: Rating annotators and obtaining cost-effective labels," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, june 2010, pp. 25–32.

[189] Y.-T. Huang, A.-J. Cheng, L.-C. Hsieh, W. Hsu, and K.-W. Chang, "Region-based landmark discovery by crowdsourcing geo-referenced photos," in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information*, ser. SIGIR '11. New York, NY, USA: ACM, 2011, pp. 1141–1142.

[190] D. C. Brabham, "Crowdsourcing as a model for problem solving: An introduction and cases," *Convergence*, vol. 14, no. 1, p. 75, 2008.

[191] W. 2.1.2005, "A lexical database of the english language," http://wordnet.princeton.edu/online/, 2011, [Online; accessed 10-September-2011].

[192] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, pp. 137–154, May 2004.

[193] B. Fitzpatrick and D. Recordon, "Thoughts on the social graph," http://bradfitz.com/social-graph-problem/, 2008, [Online; accessed 1-September-2011].

[194] M. Dean and G. Schreiber, "OWL web ontology language reference," W3C, W3C Recommendation, 2004.

[195] G. Stephen, *String searching algorithms*. World Scientific Pub Co Inc, 1994, vol. 3.

[196] P. A. V. Hall and G. R. Dowling, "Approximate string matching," *ACM Comput. Surv.*, vol. 12, pp. 381–402, December 1980.

[197] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity," in *Proceedings of the 21st national conference on Artificial intelligence - Volume 1*, ser. AAAI'06. AAAI Press, 2006, pp. 775–780.

[198] J. J. Jiang and D. W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," in *Proc. of the Int'l. Conf. on Research in Computational Linguistics*, 1997, pp. 19–33.

[199] A. ElSayed, H. Hacid, and D.-A. Zighed, "A multisource context-dependent semantic distance between concepts," in *18th International Conference on Database and Expert Systems Applications (DEXA 07), Regensburg, Germany*, ser. Lecture Notes in Computer Science, R. Wagner, N. Revell, and G. Pernul, Eds., vol. 4653. Springer, 2007, pp. 54–63.

[200] C. D. Manning and H. Schütze, *Foundations of statistical natural language processing*. Cambridge, MA, USA: MIT Press, 1999.

[201] P. D. Turney, "Mining the web for synonyms: Pmi-ir versus lsa on toefl," in *Proceedings of the 12th European Conference on Machine Learning*, ser. EMCL '01. London, UK: Springer-Verlag, 2001, pp. 491–502.

[202] G. W. Furnas, S. Deerwester, S. T. Dumais, T. K. Landauer, R. A. Harshman, L. A. Streeter, and K. E. Lochbaum, "Information retrieval using a singular value decomposition model of latent semantic structure," in *Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR '88. New York, NY, USA: ACM, 1988, pp. 465–480.

[203] H. B. Mitchell and H. B. Mitchell, "Image similarity measures," in *Image Fusion*. Springer Berlin Heidelberg, 2010, pp. 167–185.

[204] S. Cha and S. Srihari, "On measuring the distance between histograms," *Pattern Recognition*, vol. 35, no. 6, pp. 1355–1370, 2002.

[205] K. A. Baggerly, "Probability binning and testing agreement between multivariate immunofluorescence histograms: Extending the chi-squared test," *Cytometry*, vol. 45, no. 2, pp. 141–150, 2001.

[206] F. Serratosa and G. Sanromà, "A fast approximation of the earth-movers distance between multidimensional histograms," *Int. J. Patt. Recogn. Art. Intell*, vol. 22, pp. 1539–1558, 2008.

[207] A. Savakis and H. Trussell, "Blur identification by residual spectral matching," *Image Processing, IEEE Transactions on*, vol. 2, no. 2, pp. 141–151, 1993.

[208] D. Russakoff, C. Tomasi, T. Rohlfing, and C. Jr, "Image similarity using mutual information of regions," *Computer Vision-ECCV 2004*, pp. 596–607, 2004.

[209] K. Arya, P. Gupta, P. Kalra, and P. Mitra, "Image registration using robust m-estimators," *Pattern Recognition Letters*, vol. 28, no. 15, pp. 1957–1968, 2007.

[210] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar, and S. Fienberg, "Adaptive name matching in information integration," *IEEE Intelligent Systems*, vol. 18, no. 5, pp. 16–23, 2003.

[211] W. W. Cohen and E. Minkov, "A graph-search framework for associating gene identifiers with documents," *BMC Bioinformatics*, vol. 7, p. 440, 2006.

[212] A. Bilke and F. Naumann, "Schema matching using duplicates," in *21st International Conference on Data Engineering (ICDE'05)*, 2005, pp. 69–80.

[213] S. Chatzichristofis and Y. Boutalis, "Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval," in *Proceedings of the 6th international conference on Computer vision systems*. Springer-Verlag, 2008, pp. 312–322.

[214] Z. Chi, H. Yan, and T. Pham, *Fuzzy algorithms: with applications to image processing and pattern recognition*. World Scientific Pub Co Inc, 1996, vol. 10.

[215] E. Raad, B. A. Bouna, and R. Chbeir, "Bridging sensing and decision making in ambient intelligence environments," in *Multimedia Techniques for Device and Ambient Intelligence*, J. Jeong and E. Damiani, Eds. Springer US, 2009, pp. 135–164.

[216] J. Pearl, "Fusion, propagation, and structuring in belief networks," *Artif. Intell.*, vol. 29, no. 3, pp. 241–288, 1986.

[217] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.

[218] S. L. Hegarat-Mascle, I. Bloch, and D. Vidal-Madjar, "Application of dempster-shafer evidence theory to unsupervised classification in multisource remote sensing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, pp. 1018–1031, 1997.

[219] Q. Ji and M. M. Marefat, "Machine interpretation of cad data for manufacturing applications," *ACM Comput. Surv.*, vol. 29, no. 3, pp. 264–311, 1997.

[220] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *ACM SIGMOD Record*, vol. 22, pp. 207–216, June 1993.

[221] R. Baxter, P. Christen, and T. Churches, "A comparison of fast blocking methods for record linkage," in *ACM SIGKDD*, vol. 3, 2003, pp. 25–27.

[222] A. Gallagher and T. Chen, "Clothing cosegmentation for recognizing people," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, june 2008, pp. 1 –8.

[223] W. Zhang, T. Zhang, and D. Tretter, "Clothing-based person clustering in family photos," in *Image Processing (ICIP), 2010 17th IEEE International Conference on.* IEEE, 2010, pp. 4593–4596.

[224] L. L. Presti, M. Morana, and M. L. Cascia, "A data association algorithm for people re-identification in photo sequences," *Multimedia, International Symposium on*, vol. 0, pp. 318–323, 2010.

[225] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 888 –905, aug 2000.

[226] M. Thelwall, "Social network sites: Users and uses," *Advances in Computers*, vol. 76, pp. 19–73, 2009.

[227] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, "A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements," *Requirements Engineering*, vol. 16, pp. 3–32, 2011.

[228] D. Poole, "Logic, knowledge representation, and bayesian decision theory," in *CL '00: Proceedings of the First International Conference on Computational Logic*. London, UK: Springer-Verlag, 2000, pp. 70–86.

[229] S. Carberry, "Techniques for plan recognition," *User Modeling and User-Adapted Interaction*, vol. 11, no. 1-2, pp. 31–48, 2001.

[230] F. Dong, S. M. Shatz, and H. Xu, "Inference of online auction shills using dempster-shafer theory," in *ITNG '09: Proceedings of the 2009 Sixth International Conference on Information Technology: New Generations*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 908–914.

[231] H. Wu, M. Siegel, R. Stiefelhagen, and J. Yang, "Sensor fusion using dempster-shafer theory [for context-aware hci]," in *Instrumentation and Measurement Technology Conference, 2002. IMTC/2002. Proceedings of the 19th IEEE*, vol. 1, 2002, pp. 7 – 12.

[232] Y. Lu, J. Trinder, and K. Kubik, "Automatic building detection using the dempster-shafer algorithm," *Photogrammetric engineering and remote sensing*, vol. 72, no. 4, pp. 395–404, 2006.

[233] Q. J. R., "Decision trees and decision making," *IEEE trans. on systems, man and cybernetics*, vol. 20, no. 2, pp. 339–346, 1990.

[234] J. R. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[235] I. Jenhani, N. B. Amor, and Z. Elouedi, "Decision trees as possibilistic classifiers," *Int. J. Approx. Reasoning*, vol. 48, no. 3, pp. 784–807, 2008.

[236] M. J. Beynon, M. J. Peel, and Y.-C. Tang, "The application of fuzzy decision tree analysis in an exposition of the antecedents of audit fees," *Omega*, vol. 32, no. 3, pp. 231–244, June 2004.

[237] Y. Yuan and M. J. Shaw, "Induction of fuzzy decision trees," *Fuzzy Sets Syst.*, vol. 69, no. 2, pp. 125–139, 1995.

[238] L.-H. Chen and T.-W. Chiou, "A fuzzy credit-rating approach for commercial loans: a taiwan case," *Omega*, vol. 27, no. 4, pp. 407–419, August 1999.

[239] Y. Peng and P. A. Flach, "Soft discretization to enhance the continuous decision trees," in *ECML/PKDD Workshop: IDDM*, 2001, p. 109Ű118.

[240] G. J. Klir, "Where do we stand on measures of uncertainty, ambiguity, fuzziness, and the like?" *Fuzzy Sets Syst.*, vol. 24, no. 2, pp. 141–160, 1987.

[241] T. Maszczyk and W. Duch, "Comparison of shannon, renyi and tsallis entropy used in decision trees," in *ICAISC '08: Proceedings of the 9th international conference on Artificial Intelligence and Soft Computing*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 643–651.

[242] T. Wang, Z. Li, Y. Yan, and H. Chen, "A survey of fuzzy decision tree classifier methodology." in *ICFIE*, ser. Advances in Soft Computing, B. yuan Cao, Ed., vol. 40. Springer, 2007, pp. 959–968.

[243] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *ACM SIGMOD Record*, vol. 29, pp. 1–12, May 2000.

[244] M. J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, "New algorithms for fast discovery of association rules," in *3rd Intl. Conf. on Knowledge Discovery and Data Mining*. Rochester, NY, USA: University of Rochester, 1997.

[245] B. Liu, W. Hsu, and Y. Ma, "Integrating classification and association rule mining," *Knowledge discovery and data mining*, pp. 80–86, 1998.

[246] W. Li, J. Han, and J. Pei, "Cmar: accurate and efficient classification based on multiple class-association rules," in *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, 2001, pp. 369–376.

[247] X. Yin and J. Han, "Cpar: Classification based on predictive association rules," in *Proceedings of the third SIAM international conference on data mining*, vol. 3. Society for Industrial & Applied, 2003, pp. 331–335.

Appendix A

Decision Making

A.1 Bayesian Networks

Bayesian networks (BNs) [228], also known as belief networks, are directed acyclic graphs in which nodes represent domain variables, and arcs between nodes represent probabilistic dependencies [216]. These graphical structures are used to deal with uncertainty coming from different limited or noisy inputs. Bayesian networks provide a probabilistic model to deal with this uncertainty by computing a probability measure for each proposition from the causal evidence provided by its parents and the diagnostic evidence provided by its children [229]. The main limitation of Bayesian networks is that they cannot represent imprecision about uncertainty. As stated in [218] uncertainty, distinguished from imprecision, arises as a result of random process modeling or when there is missing data, where the lack of information induces some belief.

A.2 Dempster-Shafer theory

Dempster-Shafer (DS) [174] [175] is a mathematical theory of evidence that deals with uncertainty. Introduced by Dempster [174] and later extended by Shafer [175], it is used to effectively combine, accumulate and counterbalance a set of evidences from multiple data sources by employing inference mechanisms [230]. One of the advantages of Dempster-Shafer theory is that it allows to assign a probability measure with multiple possible events in contrast to the traditional probability theories that can be associated with only one possible event. The Dempster-Shafer theory may be considered as a generalization of the probability theory. As observed in [231], Dempster-Shafer theory seems closer to the way of human perception and reasoning processes than the Bayesian networks. It can be used as a data fusion algorithm that aims to improve the

quality of a decision by making use of redundant and complementary data, while decreasing its imprecision and uncertainty [232].

A.3 Decision Trees

Decision Trees (DTs) [217] are a supervised machine learning technique based on logic methods of inferring classification rules. The most well-known decision tree induction algorithms for statistical uncertainty are ID3 [233] and its successor C4.5 [234]. These algorithms use the Information Gain, a well known measure used in Decision Tree Learning, as a criterion to select the best attribute at each decision node; the best attribute is the one that partitions less randomly the objects in a data set [235]. The main characteristic of these traditional methods is that their node decisions are crisp (less than, equal, and greater than) [236]. Some works [237] [238] noted that with crisp judgments small changes in the value of the attribute may result in a sudden inappropriate change in the decision class. Others showed that there is no sharp boundaries between two satisfaction levels for some criteria in many cases [238] [239]. To overcome these shortcomings, the authors of [235] suggested a probabilistic method to construct decision trees as probabilistic classifiers. However, this suggested framework has its own limitations and inaccuracies, since the types of uncertainties arising as randomness or noise in classification problems are not necessarily to be probabilistic [237]. Dealing with real-world data means often dealing with no accurate data that contain some uncertainties. Such uncertainties, called cognitive uncertainties, can be classified into two categories: vagueness and ambiguity (described in [240]). The important development to solve cognitive uncertainty came when fuzzy techniques were introduced to soften the problem of crisp boundaries. Fuzzy techniques applied to decision trees provide a solution applicable on situations that cover cognitive uncertainty, namely vagueness and ambiguity [239]. Fuzzy decision trees, one of the most popular methods for learning and reasoning from feature-based examples, are able to improve the robustness and generalization in classification by using fuzzy reasoning. In this context, Yuan and Shaw [237] introduced an inductive fuzzy decision tree method based on the reduction of classification ambiguity with fuzzy evidence that represents classification knowledge more naturally and closer to the way of human thinking. Their fuzzy decision tree method is more robust in tolerating imprecise, conflict, and missing information. Related algorithms are one of the foundation of most large data mining packages, offering easy and computationally efficient way to extract simple decision rules [241] [242].

Appendix B

Rule Mining

B.1 Association Rule Mining

Association Rule Mining (ARM) is part of the descriptive classification category in the domain of data mining. It was introduced by [12] in order to identify relationships among a set of items in a database. To this end, it seeks to find frequent itemsets (attribute-value pair) then to derive association rules. An association rule has the form of: $X \Rightarrow Y$ where the left hand part is called the rule's body and the right hand part is called the rule's head. Let us suppose a database D of k instances having n attributes { a_1,a_2,.., a_n }. A record, d, of the database is called an itemset, and an itemset of k elements is called k-itemset. In an association rule:

X,Y { a_1,a_2,.., a_n }, Y $\neq \varnothing$ and X $\bigcap$ Y $= \varnothing$

In our case, an itemset contains always the set of attributes in addition to the defined class. The rule's head or right hand part is an attribute that represents the class type while the left hand part is the set of attributes associated to this rule's head with a specified score.

The set of interesting derived rules is obtained, from the set of all possible combinations, by measuring the interestingness of each rule. The two classical and most fundamental rule interestingness measures used are:

> **Definition 1.** ***Support:*** *The support,* s*, indicates the relative occurrence of X and Y within the overall dataset and is defined as the number of records satisfying both X and Y over the total number of records.*

It is computed as follows:

$$s(X) =\mid X \bigcup Y \mid \div k \tag{B.1}$$

Definition 2. ***Confidence:*** *The confidence,* c*, is the probability of Y given X measured by computing the ratio of the number of records satisfying both X and Yover the number of records satisfying X.*

It is computed as follows:

$$c(X \Rightarrow Y) =\mid X \bigcup Y \mid \div X \tag{B.2}$$

In the litterature, different algorithms have been proposed to derive rules such the Apriori algorithm [12], FP-Growth [243], Eclat [244]. Apriori algorithm is one of the mostly used algorithms thanks to its various advantages; first it generates frequent itemsets for a given dataset and then scans those frequent itemsets to distinguish most frequent items in this dataset. In a given dataset, association rule mining finds all rules that satisfy the minimum support and the minimum confidence. However, for association rule mining, the target of mining is not predetermined, while for classification rule mining there is one and only one pre-determined target which is the class [245].

B.2 Classification Association Rule

Classification Association Rules, or CARs, are a special subset of association rules. The head of the rules represents always the class which is not the case in the association rules. In the case of association rules, each item that doesn't appear in the body of the rule may appear in its head. This must not occur in the task of classification where only predetermined classes are selected.

This can be seen as a classification task which has been the subject of extensively researches previously [245] [246] [247]. Classification is an important part of our proposed approach that is used to discover unseen patterns and characteristics within the set of classified entities. Among the different approaches that have been proposed in the literature, we refer to: CBA [245], CMAR [246], CPAR [247]. CBA is one of the earliest and simplest algorithms for association classification. It uses the Apriori algorithm with a single rule for classification. This rule is the strongest one whose body is satisfied by the example is chosen for prediction. It organizes rules according to decreasing precedence based on confidence and then support. CMAR is similar to CBA but it uses multiple rules to perform the prediction using weighted. It uses the FP-growth algorithm for computing CARs. CPAR classifier is based on a predictive association rules and it is based on information metric. The Laplace accuracy is used to measure the accuracy of rules.

Résumé. Les réseaux sociaux occupent une place de plus en plus importante dans notre vie quotidienne et représentent une part considérable des activités sur le web. Ce succès s'explique par la diversité des services/fonctionnalités de chaque site (partage des données souvent multimédias, tagging, blogging, suggestion de contacts, etc.) incitant les utilisateurs à s'inscrire sur différents sites et ainsi à créer plusieurs réseaux sociaux pour diverses raisons (professionnelle, privée, etc.). Cependant, les outils et les sites existants proposent des fonctionnalités limitées pour identifier et organiser les types de relations ne permettant pas de, entre autres, garantir la confidentialité des utilisateurs et fournir un partage plus fin des données. Particulièrement, aucun site actuel ne propose une solution permettant d'identifier automatiquement les types de relations en tenant compte de toutes les données personnelles et/ou celles publiées. Dans cette étude, nous proposons une nouvelle approche permettant d'identifier les types de relations à travers un ou plusieurs réseaux sociaux. Notre approche est basée sur un framework orienté-utilisateur qui utilise plusieurs attributs du profil utilisateur (nom, age, adresse, photos, etc.). Pour cela, nous utilisons des règles qui s'appliquent à deux niveaux de granularité : 1) au sein d'un même réseau social pour déterminer les *relations sociales* (collègues, parents, amis, etc.) en exploitant principalement les caractéristiques des photos et leurs métadonnées, et, 2) à travers différents réseaux sociaux pour déterminer les *utilisateurs co-référents* (même personne sur plusieurs réseaux sociaux) en étant capable de considérer tous les attributs du profil auxquels des poids sont associés selon le profil de l'utilisateur et le contenu du réseau social. À chaque niveau de granularité, nous appliquons des règles de base et des règles dérivées pour identifier différents types de relations. Nous mettons en avant deux méthodologies distinctes pour générer les règles de base. Pour les relations sociales, les règles de base sont créées à partir d'un jeu de données de photos créées en utilisant le *crowdsourcing*. Pour les relations de co-référents, en utilisant tous les attributs, les règles de base sont générées à partir des paires de profils ayant des identifiants de mêmes valeurs. Quant aux règles dérivées, nous utilisons une technique de fouille de données qui prend en compte le contexte de chaque utilisateur en identifiant les règles de base fréquemment utilisées. Nous présentons notre prototype, intitulé *RelTypeFinder*, que nous avons implémenté afin de valider notre approche. Ce prototype permet de découvrir différents types de relations, générer des jeux de données synthétiques, collecter des données du web, et de générer les règles d'extraction. Nous décrivons les expériementations que nous avons menées sur des jeux de données réelles et syntéthiques. Les résultats montrent l'efficacité de notre approche à découvrir les types de relations.

Abstract. In recent years, social network sites exploded in popularity and become an important part of the online activities on the web. This success is related to the various services/functionalities provided by each site (ranging from media sharing, tagging, blogging, and mainly to online social networking) pushing users to subscribe to several sites and consequently to create several social networks for different purposes and contexts (professional, private, etc.). Nevertheless, current tools and sites provide limited functionalities to organize and identify relationship types within and across social networks which is required in several scenarios such as enforcing users' privacy, and enhancing targeted social content sharing, etc. Particularly, none of the existing social network sites provides a way to automatically identify relationship types while considering users' personal information and published data. In this work, we propose a new approach to identify relationship types among users within either a single or several social networks. We provide a user-oriented framework able to consider several features and shared data available in user profiles (e.g., name, age, interests, photos, etc.). This framework is built on a rule-based approach that operates at two levels of granularity: 1) within a single social network to discover *social relationships* (i.e., colleagues, relatives, friends, etc.) by exploiting mainly photos' features and their embedded metadata, and 2) across different social networks to discover *co-referent relationships* (same real-world persons) by considering all profiles' attributes weighted by the user profile and social network contents. At each level of granularity, we generate a set of basic and derived rules that are both used to discover relationship types. To generate basic rules, we propose two distinct methodologies. On one hand, social relationship basic rules are generated from a photo dataset constructed using *crowdsourcing*. On the other hand, using all weighted attributes, co-referent relationship basic rules are generated from the available pairs of profiles having the same unique identifier(s) attribute(s) values. To generate the derived rules, we use a mining technique that takes into account the context of users, namely by identifying frequently used valid basic rules for each user. We present here our prototype, called *RelTypeFinder*, implemented to validate our approach. It allows to discover appropriately different relationship types, generate synthetic datesets, collect web data and photo, and generate mining rules. We also describe here the sets of experiments conducted on real-world and synthetic datasets. The evaluation results demonstrate the efficiency of the proposed relationship discovery approach.

Keywords: Relationship discovery, Social networks, Rule-based relationship identification, Co-referent users, Link mining, Link type prediction, Entity resolution, Classification, Crowdsourcing, User profiles, Photos, Metadata.