Social Relationships as a Means for Identifying an Individual in Large Information Spaces

Katarína Kostková, Michal Barla, Mária Bieliková

Slovak University of Technology
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovakia
kostkova.katka@gmail.com, {barla, bielik}@fiit.stuba.sk

Abstract. In this paper we describe a method for identification of a particular user in large information spaces such as the Web is. Our main goal is to be able to decide whether the particular information found on the Web is relevant to the person being looked-up or not by taking into account additional background knowledge about that person: his or her social network. Our method combines semantically as well as syntactically based metrics to compare different social networks acquired during the identification process. We describe evaluation of the proposed method and its comparison to the related works.

Keywords: disambiguation, individual identification, social network, social networks comparison, background knowledge.

1 Introduction

Searching for information about a particular person on the Web is one of the most popular search types. However, due to the vast information space of the Web it became a non-trivial task. The problem is not only insufficient personal information on the analyzed web page, but mainly ambiguities in person search. Ambiguity exists mainly due to the fact that many persons may have the same name (multi-referent ambiguity) or many names are used in various different written forms (multi-morphic ambiguity) [13].

Our goal is to support this kind of search by automatically deciding whether a particular information found on the Web (thus a web page), is relevant to the person we are interested in or not. Our approach is not limited to be used on the Web only, but can be applied to any large information space, which suffers from name variants and name disambiguation problems. An example, of such an information space apart from the Web is DBLP[1] database, where the way of adding and representing information about new publications may cause that one author is represented by several variants of his or her name. The second, name disambiguation, problem is also present in DBLP when linking publications to their authors, i.e., how to correctly assign a publication, if there are several authors with the same name?

[1] Digital Bibliography and Library Project

Employing additional, background knowledge of the domain can solve both of the aforementioned problems. In our method the additional knowledge is represented by social networks, which identify the person by her relationships with other individuals. In reality, there is only marginal probability that two different people with the same or similar names would share the same social network. Thus social networks can be used to assign information found on the Web to a right, real world person.

In order to identify a person, we perform a comparison of two social networks, coming from different sources using both semantic (i.e., based on relations) as well as syntactic (based on text similarity) comparison metrics. The first social network is constructed automatically and is based on the data extracted from the Web. The extraction itself is driven from the starting point – a web page for which we want to decide whether it is relevant to the person we are interested in or not. The second social network comes as the person's background knowledge, e.g., from social web-based applications such as Facebook. The result of the comparison is a probability of that the input web page is related to the person of our interest.

The rest of this paper is organized as follows. In the next section, we provide an overview of related works in the field of person identification. In Section 3, we describe our method for identification of a particular user. Section 4 describes the experiments we conducted along with obtained results. Finally, we summarize the main contributions of the paper and discuss possible extensions of proposed method.

2 Related Works

The name disambiguation and name variants problems in large information spaces are in focus of many research projects. Most of approaches identify an individual in order to deal with a problem of searching for personally related information on the Web using ordinary search engines such as Google, where results are filtered, reordered or clustered appropriately [13].

During the identification process, almost every approach uses some kind of background knowledge. Some of the authors use personal information, like birth date, wife's or husband's name etc. [7], although the nature of this information imposes that it is not always possible to get it. Another option is to use keywords selected from analyzed web page [13] or estimated person's interests extracted from the web page content [10] or professional category knowledge [4]. In [2] the authors prefer to use the whole *context* of a person. By the term *context* they mean all terms extracted from the entire inspected document. The background information is used to cluster web sites found by a search engine. By assuming that the name is the only feature in common to any two namesakes, groups of pages where each group is related to one namesake are returned.

Another approach uses a social network as background knowledge [6]. The social network is created by analyzing the e-mail traffic and the content. This approach is based on an assumption that when people are connected via e-mail messages, their web pages would be connected as well. The authors issue a web search query for every name occurring in the social network, take first *K* results from each query and connect them in a graph-like structure. Pages in the most connected part of the graph are declared as pages related to a person they were interested in. The fact, that this solution is built on searching several persons at the time substantially alleviate the identification problem. However, even if the approach uses the social network, it takes only the *names* portion of it and ignores connections between people.

We believe that not only names of other people, but also relations between them are important to the identification problem and thus should be considered, which is exactly what our method does.

Most of approaches devoted to solving the name variants problems are based on dictionaries of nicknames containing pairs: name – name variant. In [3] the authors created dictionary of nicknames, based on morphological rules transforming a full name to various name variants. Another possible way is to extract nicknames from the Web. In [3] the authors also used sentence pattern "My name is <name>, friends call me <nickname>" for web-based name variants extraction. This approach allowed them to get also nicknames, which are not related to any of full name variants.

An approach to the name variants solving based on social networks can be seen in [9], where comparison of two candidates (paper authors) is performed by comparing the names of their co-authors. If an overlap of co-authors was significant, a name variant was probably found. Similar idea was elaborated in [11], where they were analyzing common publications of two distinct co-authors. If such publications could not be found (considering time and topic factors), those two co-authors are probably representing one real world person.

Most of existing works in the name disambiguation domain use some information about a person. However, the satisfactory solution of the problem of identifying a particular user was not achieved yet. We propose a method based on an analysis of social networks, which uniquely describe a person. During the processing, we do not use only the names of people from the social networks, but consider also relations between them.

3 Method of Identification of an Individual on the Web

We specify the problem as follows: "For any web page, we want to determine the probability that the web page is related to a given person". Our additional information (background knowledge) about that person is his or her social network. We propose a method, which compares two persons, represented by their names, which are either namesakes or two variants of a name of the same person. The process of the person identification consists of the following steps:

1. *Social network extraction*, where we extract social network from the person's web page, to have a second input to compare with the given background knowledge
2. *Comparison*, where the given social network is compared with information extracted in previous step.

As an input, we get a candidate web page of a person we are looking for and background knowledge (social network) related to this person. After the process of identification, we get probability of that candidate web page is related to the person we have been looking for.

3.1 Social network extraction

The extraction of social network is based on an analysis of interconnected web pages. If a relation exists between two web pages, we add a relation between people (names), which are stated on those two web pages.

The process, consisting of the following steps, is being performed recursively until the *depth of recursion* set as a parameter is not achieved:

1. An input web page URL is added to the list of URLs to be processed, the *actual depth* is set to 0.
2. If the list of URLs is empty, the process ends. Otherwise, first URL from the list is picked-up.
3. Source text of the web page is obtained for the given URL.
4. URL extraction – if the *actual depth* is lower than the given *depth of recursion*, URLs linking to other web pages are extracted from the page source and added to list of URLs to be processed. This step is omitted if the required *depth of recursion* was reached.
5. Name extraction – person names are extracted from the page source. Our approach to name extraction is based on dictionary of English given names. We search for these names on the web page and then we check whether there is a surname, middle name, or middle name initial around the found given name, all using regular expressions.
6. Social network creation – names and relations between web pages are added to social network.
7. Increment the *actual depth* and continue with step 2.

In step 4 we extract references to other websites. By following links between web pages the relations between persons are obtained.

3.2 Social network comparison

In second step of our method we compare two social networks, the one given as background knowledge and the one extracted from the Web, by examining their nodes and edges. Each node represents one person. We do not merge the two networks, but rather perform a mapping of the nodes based on syntactical similarity (in our case, we decided to use Levenshtein distance) of the node names. The mapping produces one-to-one as well as one-to-many connections between the two networks (Fig. 1). These connections are subsequently used to compare the networks.

However, by employing only syntactic comparison, we achieve unsatisfactory results, with many name variants still unresolved. Therefore we use syntactically based metric only as a pre-filter to a semantic-based comparison (which is thus performed only for nodes, which have the syntactic similarity above the pre-defined threshold). The threshold differs according to the nature of the social networks to be compared. For example, if we have a social network of authors from DBLP, proper similarity of names can be higher than in a social network from some chat portal.

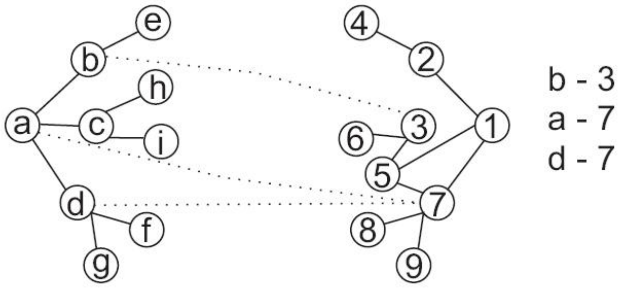

Fig. 1. Syntactic mapping of two social networks

We employed existing semantic metric based on relations *Connected Triple* [12] for comparison. When social network is defined as a graph G with vertices V representing people and edges E representing relations between them, $G(V,E)$, then a connected triple is a graph $G^{'}(V_{CT},E_{CT})$ of three vertices with $V_{CT}=\{X,Y,Z\}$, $V_{CT}\subset V$ and two edges $E_{CT}=\{E_{XY},E_{YZ}\}$, $E_{CT}\subset E$, where $E_{XZ}\not\subset E$. On Figure 2, we present an example of a connected triple – the two vertices (persons) being compared must not be connected by an edge. This is obvious, as if there had been an edge between two compared vertices, these two persons would have knew each other or would have worked together, which means they are *really* two different persons. If the edge between them is missing, then there is a possibility that these two vertices represent the *same person*.

The process resulting in similarities of pairs of vertices from two social networks consists of following steps:

1. Select a pair of vertices, each coming from different social network.
2. Compute similarity of vertices using Levenshtein distance
   a. If the similarity is above the threshold
      i. compare the vertices using enhanced Connected Triples
   b. If the similarity is below the threshold
      i. continue with another pair of vertices
      ii. finish if there are no more unprocessed pairs

The basic original formula for calculating similarity between two persons based on connected triples was defined in [12]:

$$similarity(i,j)=\frac{\left|C_{ij}\right|}{\max_{k=0..m,l=0..n}\left|C_{kl}\right|} \tag{1}$$

where i a *j* are compared persons*,* C_{ij} is a count of connected triples of compared persons and C_{kl} is a count of connected triples where at least one the compared persons is involved.

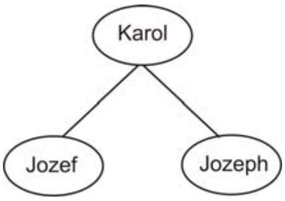

Fig. 2: A connected triple. Note, that there is no relation between Jozef and Jozeph.

The similarity is computed as a number of connected triples between compared vertices, divided by a maximum of connected triples in graph, between any two vertices. Notice, that this formula considers all relations in the graph, even those that are not related to the people being compared.

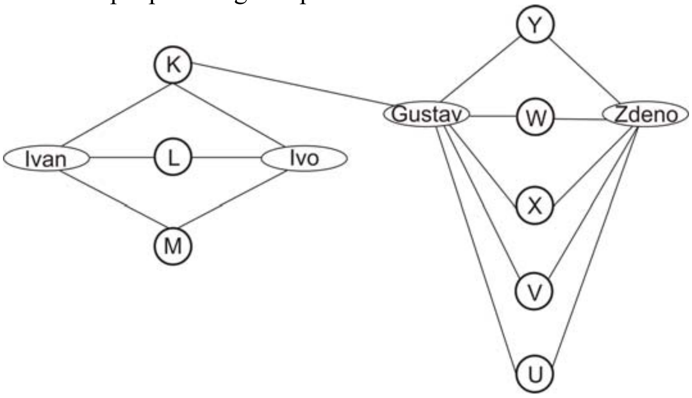

Fig. 3: An example of social network.

Let us consider a social network depicted in Figure 3, where we are comparing two persons, Ivan and Ivo, and we know that it is the same person. If we use formula 1, the probability that these two vertices represent the same person in reality is 3/5, because number of connected triples where both members are the nodes being compared is 3, but maximum number of connected triples between any two nodes is 5, between *Gustav* and *Zdeno*.

We modified the formula 1 considering only relations and connected triples, where at least one of its members is one of the compared persons. In other words, we count similarity of two people as a number of connected triples, where members are both of them, divided by a maximum number of connected triples, where at least one of them is a member. The new formula with the same meaning of variables as in formula 1:

$$similarity(i, j) = \frac{|C_{ij}|}{\max_{k=i \vee j, l=0..n} |C_{kl}|} \qquad (2)$$

Taking the same example of social network from Figure 3, the probability that Ivan and Ivo is the same person will be 3/3, because the number of connected triple where both members are compared nodes is 3 and also maximum number of connected triple, where at least one member is one of the compared nodes is 3.

The connected triples are detected according to the following algorithm

1. take the first relationship from the first social network in the form *"from node A to node B"*
2. consider node *A* as an intermediate node
    a. find a mapping of node A in the second social network (node 3 from the example in Fig. 4)
    b. find all adjacent of mapped node in the network (nodes 8,9,1)
    c. add corresponding triples (B,A,x) into connected triples, where x is the found adjacent node from the second network
3. consider node *B* as an intermediate node and perform the same procedure as in step 2

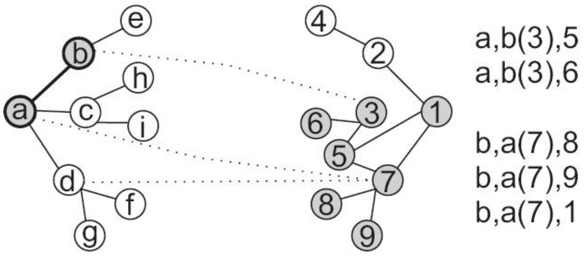

Fig. 4. Identification of connected triples between two social networks

After having identified the connected triples, we can proceed to the comparison itself based on the aforementioned formula 2.

4 Evaluation

We developed a software tool, which implements the aforementioned method in order to evaluate its properties. We evaluated name extraction, used during the social network construction based on linked web pages as was described in section 3.1 and our method for social networks comparison based on syntactic and semantic properties of the networks.

4.1 Name extraction

We evaluated our name extractor on real web pages, from which we extracted names and then manually checked the results in order to determine precision and recall of our approach. We took pages of American universities and homepages of professors from these universities, usually containing lists of their students and/or publications. We did not consider any structural features of the pages (HTML markup) and were extracting names in different combinations of first name, middle name, middle name initial and surname from the full texts. Table 1 shows our results.

We achieved a satisfactory precision, but low recall. The reason of lower recall values is that we used a dictionary of English names, but our dataset contained several pages with many foreign names, like Bernd, Xuerui or Sameer which were not included in the dictionary. Using extended dictionary of first names can solve this problem.

Table 1: Name extraction results

	Mean	Standard Deviation	Min	Max
Precision	95,8%	2,6	94,7%	98,4%
Recall	51,3%	12,2	31,6%	59,8%

4.2 Connected triples

We evaluated our approach to social network comparison based on the modified connected triple method. We obtained a social network from Slovak Companies Register on the Internet (http://orsr.sk), which has about 300 000 vertices and more than 460 000 edges, representing companies and individuals, which are somehow connected to those companies. For prototyping purposes, we decided to take only a part of this social network.

We have selected all people with surname “Havran” and all relations of these people. We did the same for surname “Novak” as these two surnames are very common in Slovakia. In order to evaluate precision and recall of our approach, we identified duplicities in that smaller social networks manually as well as by using a domain-specific heuristics (baseline) employing specific methods to identify duplicates in the whole social network. The duplicates, in this case, were persons with very similar names and addresses.

The results are shown in Table 2. We achieved good results in precision and recall and we also found more duplicates than the baseline method. However, we did not find all duplicates, which the baseline did, which can be caused by the fact that we took only a part of the whole social network whereas the baseline operated on the whole graph. We also compared our modified connected triple algorithm with an original connected triple algorithm. Results are shown in Table 3.

Table 2: Connected triple algorithm results

	Havrans	Novaks
Vertices	329	256
Edges	610	526
Levenshtein 85% and connected triples (our method)	28	23
Levenshtein 85% and connected triples and same address	23	15
Baseline	14	19
Our method ∩ baseline	9	12
Precision	82,2%	65,2%
Recall	82,2%	60,0%

Table 3: Modified connected triple algorithm vs. original connected triple algorithm

	Havrans	Novaks
Identified duplicates with similarity > 50% (by our modified method)	9	20
Identified duplicates with similarity > 50% (by the original method)	3	0
Precision of modified	- 23,3%	+ 56,5%
Recall of modified	+ 14,3%	+ 52,5%

When we defined the required similarity of duplicates to be more than 50 %, we got better recall values with our modified algorithm and even if precision achieved on Havrans dataset is lower, we identified more duplicates than the original connected triple algorithm. We can thus conclude that our modification was useful.

5 Conclusions

In this paper we presented a novel method for identification of a particular user in the large information space such as the Web. We based our solution on social networks, which act as a background knowledge about the person. We combine syntactically and semantically based metrics in the process of network comparison, which determines whether the information we found on the Web is related to the person we are looking for or not.

We evaluated our approach to gain verification of our changes in the original connected triple metric as well as to verify our assumptions about overall usability of our method against specialized, domain dependent heuristics. The experiments showed that our *domain independent approach* based on a modified connected triple metric is performing well, compared to either domain dependent heuristics or original connected triple metric. More, we were able to discover *different* duplicities than the domain dependent heuristics, which promises that their combination would allow for the achievement of even better results as they eliminate the weak points of each other.

We should point out that our method is built on a broader concept of social networks comparison, which has a great usage potential especially in, but not limited to, the domain of user modeling [1]. We can determine relevance of data found on the Web, compare two social networks and find corresponding nodes within them. We see a nice application of our method in social portals. When a new user registers into a social portal, we can ask for his homepage (or any other page stating information which is relevant to that user). We then extract his social network from the “web environment” of this page and take it as our background knowledge about this new user. Subsequently, we can use it to find his friends, which are already registered within our social portal. Then we can recommend these people to the newly registered user as potential friends or we could offer him services, which his friends are also interested in.

Our method of identification of an individual can also serve to other approaches such as query expansion based on user context [8], where it can help to disambiguate entities present in the current user context. Apart from search-related tasks, it can also assist in automatic content annotation [5], where it helps to associate a correct instance to a retrieved information.

We already see several possible extensions of our work. Probably the most interesting is to consider different attributes of people or relations in the algorithm. For instance, different types of relationships may have different weights of contribution to the final result, depending on their importance. Promising seems to be also a combination of our approach with various clustering methods and techniques.

Acknowledgement: This work was partially supported by VEGA 1/0508/09, KEGA 028-025STU-4/2010 and it is the partial result of the Research & Development Operational Programme for the project Support of Center of Excellence for Smart Technologies, Systems and Services, ITMS 26240120029, co-funded by the ERDF.

References

1. Barla,M. et al.: Rule-based User Characteristics Acquisition from Logs with Semantics for Personalized Web-based Systems. In Computing & Informatics, Vol. 28, No. 4, 2009, pp. 399-427,.
2. Bollegala, D., et al.: Disambiguating Personal Names on the Web Using Automatically Extracted Key Phrases. In: ECAI 2006, IOS Press, 2006, pp. 553–557.
3. Driscoll, P., Yarowsky, D.: Disambiguation of Standardized Personal Name Variants, 2007, available online: www-lipn.univ-paris13.fr/~poibeau/mmies/mmies04.pdf [17.05.2010].
4. Han, X. and Zhao, J. Web personal name disambiguation based on reference entity tables mined from the web. In Proc. of the 11th Int. Workshop on Web information and Data Management, Hong Kong, China, 2009. WIDM '09. ACM, New York, NY, pp. 75-82.
5. Laclavík, M., et al.: Ontea: Platform for Pattern Based Automated Semantic Annotation. Computing and Informatics, Vol. 28, 2009, No. 4, pp. 555–579.
6. Malin, B.: Unsupervised Name Disambiguation via Social Network Similarity. In: Workshop on Link Analysis, Counterterrorism, and Security, 2005.
7. Mann, G.S., Yarowsky, D.: Unsupervised Personal Name Disambiguation. In: Natural Language Learning Workshop at HLT-NAACL 2003, Association for Computational Linguistics, 2003, pp. 33–40.
8. Návrat, P. Taraba, T. Bou Ezzeddine, A., Chudá, D. Context Search Enhanced by Readability Index. In: IFIP Series. Vol. 276: Artificial Intelligence in Theory and Practice. IFIP WCC, 2008, New York : Springer Science+Business Media, LLC. pp. 373-382.
9. On, B., Lee, D., Kang, J., and Mitra, P.: Comparative study of name disambiguation problem using a scalable blocking-based framework. In Proc. of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '05. ACM, New York, NY, 2005, pp. 344-353.
10. Omelina, Ľ: Extracting Information from Web Pages Based on Graph Models, In *IIT.SRC'09: Proc. of Student Research Conference*, Bieliková, M. (Ed.), STU Bratislava, 2009, pp. 105-112.
11. Reuther P.: Personal Name Matching: New Test Collections and a Social Network based Approach., University of Trier, Mathematics/Computer Science: Tech. Report 06-01, 2006.
12. Reuther, P., et al.: Managing the Quality of Person Names in DBLP. In: ECDL'06. LNCS 4172, Springer, 2006, pp. 508-511.
13. Wan, X., Gao, J., Li, M., and Ding, B.: Person resolution in person search results: WebHawk. In Proc. of the 14th ACM Int. Conf. on Information and Knowledge Management, Bremen, Germany, 2005. CIKM '05. ACM, New York, NY, pp. 163-170.