CASIANED: Web Personal Name Disambiguation Based on Professional Categorization

Xianpei Han
National Laboratory of Pattern Recognition
Zhongguancun East Road 95
HaiDian District, Beijing, China, 100190
+86 010 82614468
xphan@nlpr.ia.ac.cn

Jun Zhao
National Laboratory of Pattern Recognition
Zhongguancun East Road 95
HaiDian District, Beijing, China, 100190
+86 010 82614505
jzhao@nlpr.ia.ac.cn

ABSTRACT

In this paper, we describe the web personal name disambiguation system of the CASIANED team for the second Web People search evaluation (WePS-2). Based on the assumption that professional category information can be used to distinguish the namesakes from each other, we disambiguate personal names by categorize them into a real world professional taxonomy which is extracted from Freebase. For every ambiguous personal name, firstly we detect its target professions from the professional taxonomy and extract the training sets for every detected profession through web mining. Then the personal name appearances are categorized into the detected professions using a kNN classifier trained using the above training sets. The personal name appearances being categorized into the same professional category will be clustered into a single cluster referred to the same individual. The experimental results show that our system can achieve robust performance on different datasets.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information storage and retrieval–*Information Search and Retrieval.*

General Terms

Algorithms, Experimentation

Keywords

Name Disambiguation, Person Resolution, Web Person Search

1. Introduction

Web person search is one of the most frequent search types on the web search engine. However, in response to a personal name query, search engines usually return a long, flat list of search results containing web pages about several namesakes. For example, the Google search results of "*Michael Jordan*" contain more than ten namesakes, some examples are shown below:

1. *Michael (Jeffrey) Jordan, Basketball Player*
2. *Michael Jordan, Football Player*
3. *Michael (I.) Jordan, Professor of Berkeley*
4. *Michael (B.) Jordan, American actor*

The effectiveness of web person search could be greatly improved if the search results can be grouped according to their referents. On the other hand, an ever-increasing number of question answering, information extraction systems are coming to rely on the data on the web, ambiguous names will lead to wrong answers and poor results. A disambiguation step will help these systems to achieve better results.

The goal of web personal name disambiguation, as defined in the clustering task guideline of WePS [1][14], is to cluster the personal name appearances within different web pages according to their referents. Thus, a disambiguation system receives a set of search results from a person search scenario, and outputs a clustering of these results, where each cluster is assumed to contain all (and only those) the pages that refer to the same individual.

Previous research on name disambiguation mainly employs clustering algorithms which disambiguates ambiguous names in a given document collection through clustering them into different reference entities [5][6][7][8]. The clustering method firstly computes the similarities between different name appearances, then clusters them using a pre-defined or trained end condition. There have been two types of clustering-based disambiguation methods. One clustering method disambiguates personal names based on their context similarity. Bagga and Baldwin (1998) represent a name as a vector of its context words, the similarity between two names were determined by the co-occurring words, then two names were predicted to be the same entity if their similarity scores are above a threshold[5]. Mann and Yarowsky (2003) extended the name's vector presentation by extracting the structured biographic facts, such as the birth day, birth year and occupation [6]. Ted Pedersen et al. (2004) employed significant bigrams to represent the contexts of a name[8]. Fleischman (2004) trained a Maximum Entropy model to give the probability that two names refer to the same reference entity, then a modified agglomerative clustering algorithm was used to cluster names using the probability as the similarity[7]. Another clustering method is the graph-based method, which computes the similarity using the relation information or the link structure in a social network. Bekkerman and McCallum (2005) disambiguated names based on link structure of web pages, their model leverages hyperlinks and the distance between pages [10]. Malin and Airoldi (2005) and Malin (2005) measured the similarity based on the probability of walking from one ambiguous name to another in a random walk of the social network constructed from all documents[9][11]. Minkov et al. (2006) disambiguated names in email documents by measuring the similarity between documents

WWW 2009, April 20-24, 2009, Madrid, Spain.

and other objects in the graphs built from the email data, a lazy graph walk is used to compute the similarity[2].

The previous researches on clustering methods were focused on choosing a better similarity measure. However, it is also a remaining question to determine the optimal parameter setting, especially in an open environment such as on the web [13]. Another problem is that a special step is needed to generate an informative description for each cluster [12].

In order to address these problems, the CASIANED system disambiguates personal names based on professional categorization. The starting point of our method is that professional category information can be used to disambiguate personal name appearances. So given a personal name, instead of clustering personal name appearances according to their local similarity, our system categorizes different personal name appearances into a real world professional taxonomy. The personal name appearances that are categorized into the same professional category will be clustered into a single cluster.

This paper proceeds as follows. Section 2 describes our proposed system in detail. Section 3 includes the performance results and discussions. This paper concludes with a review of summary and future directions.

2. Our Method

In this section, we describe our web personal name disambiguation system. The starting point of our method is that professional category information can be used to distinguish namesakes from each other. For example, "*Michael Jordan*" can refer to several persons, we can distinguish them using "*Michael Jordan (Basketball player)*", "*Michael Jordan (Football player)*", "*Michael Jordan (Politician)*", etc. So if an appropriate professional taxonomy is given, we can disambiguate personal name appearances by linking it with an appropriate professional category. In this paper, we disambiguate personal name appearances by categorizing them into a professional taxonomy extracted from Freebase.

A graphical diagram presenting our method is shown in Figure 1. The method is essentially composed of two function modules: the profession detection module and the professional categorization module. Given a personal name, the profession detection module detects the target professions that a given personal name may be categorized into, then the professional categorization module disambiguates the ambiguous personal name appearances by categorizing them into the detected professions. A more detailed discussion for each module will be presented in the following subsections.

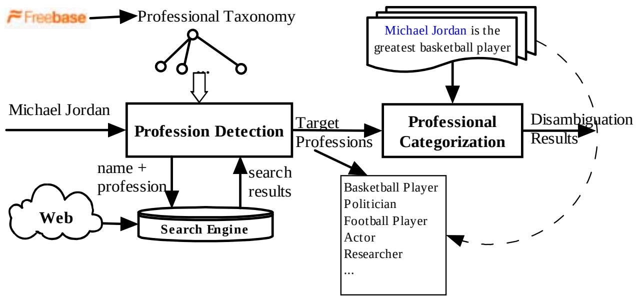


Figure 1. An abstract diagram presenting the main components of CASIANED system

2.1 Profession Detection

Usually a real world professional taxonomy contains thousands of professional categories, but a personal name usually only corresponds to a small set in the whole professional categories. So using the whole taxonomy as the categorization target will lead to high error-rate and noises. So we employ an external step to detect only the professions that the given personal name may really corresponding to, which we call target professions in this paper.

The profession detection module detects the target professions of a personal name within a professional taxonomy. For example, for the name "*Michael Jordan*", we detect the target professions as basketball player, politician, football player, etc.

The profession detection module detects the professions using a knowledge based web mining method. We first extract a real world professional taxonomy from the social database web site Freebase[1]. Based on the extracted professional taxonomy, we mine the web through heuristic search for finding the evidence that a profession is the target profession of a personal name. In the end, all professions whose evidence can be found on web will be retained as the target professions.

2.1.1 Professional Taxonomy Extraction

As described above, the professional taxonomy is the base to support our disambiguation system. The professional taxonomy should be large enough so that it can cover most of the professions in the real world. And it should be high-quality so that it will not contain too much noisy information.

We extract a real world professional taxonomy from the Freebase. The Freebase is a social database which provides structured data in different domains, and it includes a People dataset[2] which contains more than 800 thousands of persons labeled with more than 2,000 professional categories. It is large enough to cover most of the professions in the real world. On the other hand, most of the data on the Freebase are manually edited, so it is high-quality also.

We extract the professional taxonomy through the following processes. Firstly, all the professional categories in the Freebase are extracted, then we manually eliminate the professional categories which are too general (for example, "person") or too specific (for example, "Mayor of Wellington"). Finally, we extract 1,712 professional categories.

2.1.2 Evidence Finding through Web Mining

In this section we try to find the evidence for every profession within the extracted professional taxonomy can be used as the target profession of the given personal name. For example, for the personal name "Michael Jordan", we try to find the evidence that the "basketball player" can be used as its target profession.

On the web, the evidence about a profession can be used as a given personal name's target profession is the web pages talked about the person who both have the given personal name and the profession, which we called *evidence pages* in this paper. For example, the Wikipedia page *http://en.wikipedia.org/wiki/Michael_Jordan* is an *evidence page* for the profession "*basketball player*" to be the target profession of "*Michael Jordan*". In this paper, we define an evidence page as:

[1] http://www.freebase.com/

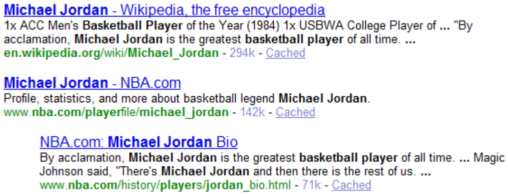
[2] http://www.freebase.com/view/people/

*A web page is an **evidence page** if there is at least one sentence which contains both the name and the profession in this web page.*

The *evidence pages* can be found through web search. We illustrate this process using an example, which finds evidence pages of the profession *basketball player* can be used as the target profession of the personal name *Michael Jordan*:

1) We build a search query using the name together with the profession keyword, i.e. *"Michael Jordan"+"basketball player"*.

2) The query is submitted to a search engine and the returned results are processed to find the evidence pages. For the above example, the first and the third search result of the top 3 Yahoo search results, which is shown in Figure 2, are confirmed to be *evidence pages*.



Michael Jordan - Wikipedia, the free encyclopedia
1x ACC Men's **Basketball Player** of the Year (1984) 1x USBWA College Player of **...** "By acclamation, **Michael Jordan** is the greatest **basketball player** of all time. **...**
en.wikipedia.org/wiki/**Michael_Jordan** - 294k - Cached

Michael Jordan - NBA.com
Profile, statistics, and more about basketball legend **Michael Jordan**.
www.**nba.com**/**player**file/**michael_jordan** - 142k - Cached

NBA.com: **Michael Jordan** Bio
By acclamation, **Michael Jordan** is the greatest **basketball player** of all time. **...** Magic Johnson said, "There's **Michael Jordan** and then there is the rest of us. **...**
www.**nba.com**/history/**players**/**jordan_**bio.html - 71k - Cached

Figure 2. The Top 3 search results of *"Michael Jordan"* +*"basketball player"* using *Yahoo* search engine

After the evidence page searching process, an evidence page count threshold is then set to filter these professions. The retained professions will be used as the target professions of the given personal name.

2.2 Professional Categorization using kNN Classifier

We disambiguate personal name appearances by categorizing them into the detected professions. We assume that the same names in the same web page will refer to the same individual. Thus the task is to category the web pages. The professional categorization process is composed of three steps: 1) To build the training set for every profession within the detected professions; 2) To extract the feature representation for a web page; 3) To choose a proper classification algorithm. The detailed description for every step is as follows.

2.2.1 To build the training set every profession

For every profession within the detected professions, we need to build a training set to train the classifier. As the evidence web pages always provide the information about the profession of the given personal name, we use the evidence page set of a profession as its training set.

2.2.2 To extract the feature representation for a web page

We need to extract the representation of a web page as the input of the categorization process. A web page is represented as a vector of features as follows.

Tokens. Identical to the systems in Javier Artiles et al. (2005) [4], we segment the web page's text content into words and then stem them using the Porter stemmer[3], stop words are filtered. Each word retained is used as a feature and weighted by its Term Frequency× Inverse Document Frequency (TF×IDF).

Snippet Tokens. We also tokenize the snippet text using the same method as web page's text. All tokens within the snippet will be used as feature.

Named Entities. We extract the named entities from the web page using the OpenNLP(http://opennlp.sourceforge.net/) Named Entity Detection tools. This tagger identifies and labels names of places, organizations, people, time and date in the input text. Each named entity is treated as a feature and weighted by TF×IDF, too.

URL Tokens. Not only the web page contains rich information, its URL also contains rich information. For example, the URL *http://en.wikipedia.org/wiki/Michael_Jordan_(footballer)* indicates that the *Michael Jordan* mentioned in this web page is a footballer. We segment the URL into tokens and filter the common URL part such as *http*, *www*, *org*, etc. All retained tokens are treated as a feature and is TF×IDF weighted.

2.2.3 The kNN Classifier

Given the list of detected professions (p_1, p_2, …, p_m), a web page *wp* is classified into a specific profession *p* using the kNN classifier [3] according to the formula below:

$$p = \arg\max_{p_i} Similarity(p_i, wp)$$

where the similarity between a specific profession in the reference personal entity list and a web page is determined by the max cosine similarity between the web pages in the training set of the profession and the web page to be classified:

$$Similarity(p_i, wp) = \max_{ep \in p_i.EP} Co\sin e(ep, wp)$$

After the professional categorization process, the web pages being classified into the same professional category will be clustered into a single cluster, and the cluster is represented by the personal name and the profession, such as (*Michael Jordan, basketball player*).

3. Result and Discussion

To assess the performance of our method and compare it with the traditional methods, we conduct a series of experiments. We experiment our system on the WePS1_training dataset, the WePS1_test dataset, and the final test data set of WePS2. The official run results for WePS2 is shown in Table 1, while the five runs are corresponding to different evidence page count thresholds. We also compared our method with the traditional clustering-based disambiguation methods.

Table 1. Official Run Results for WePS2

Runs	BEP	BER	F-0.5
CASIANED_1	0.70	0.62	0.60
CASIANED_2	0.76	0.60	0.57
CASIANED_3	0.88	0.45	0.51
CASIANED_4	0.65	0.75	0.63
CASIANED_5	0.60	0.81	0.63

In order to compare our method with the traditional clustering methods, we implement two clustering-based baselines: the Clustering_Simi and the Clustering_K. Both two baselines cluster web pages using agglomerative clustering algorithm with single linkage, but the Clustering_Simi uses the similarity threshold as the end condition and the Clustering_K uses the cluster number as the end condition. For every dataset, we train the end condition using the other two datasets, so two clustering results can be obtained for each dataset.

[3] http://tartarus.org/~martin/PorterStemmer/

The results of web name disambiguation are shown in Table 2. As shown in these results, our system achieves relative good results and the F-measures of our CASIANED system on the three datasets are very close, i.e. 0.66, 0.65, 0.63 seperately .

Based on the experimental results shown in Table 2, we can make the following observations:

1. Web personal name disambiguation is necessary, for the two none disambiguation baselines, i.e. **All-in-ONE** and **ONE-in-ONE**, perform poorly.
2. It is difficult to choose the optimal clustering parameters that can achieve robust disambiguation performance on various personal names on the web. As shown in Table 2, the performance fluctuations of the two clustering baselines are significant in different data sets and in different end conditions. For example, the Clustering_Simi can achieve the F-measure 0.71 on the WePS1_training dataset, but the performance declines to 0.34 on the WePS1_test dataset.
3. Compared with the clustering methods, our method achieves more robust results on the web. As shown in Table 2, the performance fluctuation of our method is very slight, while the performance fluctuations of the clustering methods are significant.

Table 2. The web personal name disambiguation results

Method	**WePS1_training**		
	BEP	**BER**	**F-0.5**
All_in_ONE	0.54	1.0	0.64
ONE_in_ONE	1.0	0.39	0.47
Clustering_Simi	0.58 0.63	0.96 0.91	0.68 0.71
Clustering_K	0.91 0.87	0.48 0.57	0.56 0.62
CASIANED	**0.77**	**0.71**	**0.66**
	WePS1_test		
	BEP	**BER**	**F-0.5**
All_in_ONE	0.17	1.0	0.25
ONE_in_ONE	1.0	0.43	0.57
Clustering_Simi	0.22 0.24	0.97 0.97	0.32 0.34
Clustering_K	0.59 0.27	0.81 0.95	0.64 0.38
CASIANED	**0.66**	**0.74**	**0.65**
	WePS2_test		
	BEP	**BER**	**F-0.5**
All_in_ONE	0.43	1.0	0.53
ONE_in_ONE	1.0	0.24	0.34
Clustering_Simi	0.47 0.57	0.96 0.88	0.56 0.63
Clustering_K	0.47 0.76	0.95 0.60	0.56 0.60
CASIANED	**0.65**	**0.75**	**0.63**

4. Conclusion

This paper describes our web personal name disambiguation system for WePS-2 evaluation. Based on a real world professional taxonomy extracted from Freebase, we disambiguate the personal name appearances by categorizing them into appropriate professions. The method is unsupervised – it only involves human efforts in choosing professional category knowledge.

Our method achieves appealing performance for web personal name disambiguation: compared with the clustering methods, our method can achieve more robust results. The disambiguation method used in this paper can also be used in disambiguating other types' entities by selecting an appropriate taxonomy.

5. ACKNOWLEDGMENTS

The work is supported by the National Natural Science Foundation of China under Grants no. 60673042, 60875041 and the National High Technology Development 863 Program of China under Grants no. 2006AA01Z144.

6. REFERENCES

[1] Javier Artiles, Julio Gonzalo, Satoshi Sekine. The SemEval-2007 WePS Evaluation: Establishing a benchmark for the Web People Search Task. SemEval-2007.

[2] Einat Minkov, William W. Cohen, Andrew Y. Ng. Contextual Search and Name Disambiguation in Email Using Graphs. In Proceedings of SIGIR, 2006.

[3] DASARATHY,B.V. Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. McGraw-Hill Computer Science IEEE Computer Society Press, Los Alamitos, California. 1991.

[4] Javier Artiles, Julio Gonzalo, and Felisa Verdejo. A testbed for people searching strategies in the WWW. In Proceedings of SIGIR, 2005.

[5] Amit Bagga, Breck Baldwin. Entity-Based Cross-Document Coreferencing Using the Vector Space Model, In Proceedings of HLT/ACL, 1998.

[6] Gideon S. Mann and David Yarowsky. Unsupervised Personal Name Disambiguation. In Proceedings of CONIL, 2003.

[7] Michael Ben Fleischman. Multi-Document Person Name Resolution, In Proceedings of ACL, 2004.

[8] Ted Pedersen, Amruta Purandare, Anagha Kulkarni. Name Discrimination by Clustering Similar Contexts. In Proceedings of CICLing, 2005.

[9] Bradley Malin, Edoardo Airoldi. A Network Analysis Model for Disambiguation of Names in Lists. In Proceedings of CMOT, 2005

[10] Ron Bekkerman, Andrew McCallum. Disambiguating Web Appearances of People in a Social Network. In Proceedings of WWW, 2005

[11] Bradley Malin. Unsupervised Name Disambiguation via Social Network Similarity, In Proceedings of SIAM, 2005.

[12] Xiaojun Wan, Jianfeng Gao, Mu Li, Binggong Ding. Person Resolution in Person Search Results: WebHawk. In Proceedings of CIKM, 2005.

[13] Ying Chen, James Martin. Towards Robust Unsupervised Personal Name Disambiguation. In Proceedings of EMNLP, 2007.

[14] Javier Artiles, Julio Gonzalo and Satoshi Sekine. WePS2 Evaluation Campaign: overview of the Web People Search Clustering Task. In 2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference, 2009.