PRIS at Chinese Language Processing

--Chinese Personal Name Disambiguation

Jiayue Zhang, Yichao Cai, Si Li, Weiran Xu, Jun Guo
School of Information and Communication Engineering
Beijing Universit of Posts and Telecommunications
jyz0706@gmail.com

Abstract

The more Chinese language materials come out, the more we have to focus on the "same personal name" problem. In our personal name disambiguation system, the hierarchical agglomerative clustering is applied, and named entity is used as feature for document similarity calculation. We propose a two-stage strategy in which the first stage involves word segmentation and named entity recognition (NER) for feature extraction, and the second stage focuses on clustering.

1 Introduction

World Wide Web (WWW) search engines have become widely used in recent years to retrieve information about real-world entities such as people. Web person search is one of the most frequent search types on the web search engine. As the sheer amount of web information expands at an ever more rapid pace, the named-entity ambiguity problem becomes more and more serious in many fields, such as information integration, cross-document co-reference, and question answering. It is crucial to develop methodologies that can efficiently disambiguate the ambiguous names form any given set of data.

There have been two recent Web People Search (WePS) evaluation campaigns [1] on personal name disambiguation using data from English language web pages. Previous researches on name disambiguation mainly employ clustering algorithms which disambiguates ambiguous names in a given document collection through clustering them into different reference entities. However, Chinese personal name disambiguation is potentially more challenging due to the need for word segmentation, which could introduce errors that can in large part be avoided in the English task.

There are four tasks in Chinese Language Processing of the CIPS-SIGHAN Joint Conference, and we participate in the Chinese Personal Name Disambiguation task. To accomplish this task, we focused on solving two main problems which are word segmentation and duplicate names distinguishment. To distinguish duplicate names, the system adopts named entity recognition and clustering strategy. For word segmentation and NER, we applied a sharing platform named LTP designed by Harbin Institute of Technology [2].This tagger identifies and labels names of locations, organizations, people, time, date, numbers and proper nouns in the input text.

The paper is organized as follows. Section 2 introduces our feature extractions along with their corresponding similarity matrix learning. In Section 3, we analyze the performance of our system. Finally, we draw some conclusions.

2 Methodology

Our approach follows a common architecture for named-entity disambiguation: the detection of ambiguous objects, feature extractions and their corresponding similarity matrix learning, and clustering. The framework of overall processing is shown in Figure 1.

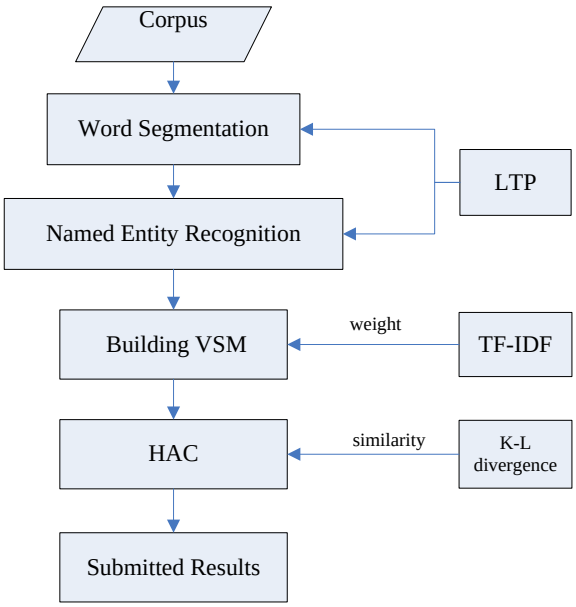

Figure 1. System Framework

2.1 The detection of ambiguous objects

Since it is common for a single document to contain one or more mentions of the ambiguous personal name, that is to say, the personal name may appear several times in one document, there is a need to define the object to be disambiguated. Here, we adopt the policy of "one person per document" (all mentions of the ambiguous personal name in one document are assumed to refer to the same personal entity in reality) as in [3] [4] [5]. Therefore, an object is defined as a single entity with the ambiguous personal name in a given document. This definition of the object (document-level object) might be not comprehensive, because the mentions of the ambiguous personal name in a document may refer to multiple entities, but we found that this is a rare case (most of those cases occur in genealogy web pages). On the other hand, the document-level object can include much information derived from that document, so that it can be represented by features [6].

For a given ambiguous personal name, word segmentation is applied first. Then we try to extract all mentions of the ambiguous personal name. Take the given personal name "高军" for example, first, the exact match of the name is extracted. Secondly, mentions that are superstrings of the given name like "高军田"is also extracted . Finally , mentions that contain character sequences but not a personal name like "最高军事法院" is ignored.

Given this definition of an object, we define a target entity as an entity that includes a mention of the ambiguous personal name.

2.2 Feature extraction and similarity matrix learning

Most of the previous work ([3] [4] [5]) used token information in the given documents. In this paper, we follow and extend their work especially for a web corpus. Furthermore, compared to a token, a phrase contains more information for named-entity disambiguation. Therefore, we explore both token and phrase-based information in this paper. Finally, there are two kinds of feature vectors developed in our system, token-based and phrase-based. The token-based feature vector is composed of tokens, and the phrase-based feature is composed of phrases. The two feature vectors are combined into a unified feature vector in which tf-idf strategy is used for similarity calculation.

2.2.1 Named Entity Features

From the results and papers of various teams participating WePS, NEs have been shown to be effective features in person name disambiguation, so we used NEs as features in this study. Through observation, we found that two different individuals can be identified by their corresponding NEs, especially by location, organization name and some proper nouns. Hence, in our study, we only extracted person, location, organization name and proper noun as feature from the output of LTP, while time, date and numbers are discarded. However, location and organization name have many proper nouns related weakly to a certain person. Therefore, terms having high-document-frequency in training data sets are removed from test data.

2.2.2 Similarity matrix learning

After NE extraction, we applied the vector space model to the calculation of similarities between

features. In the model, tf-idf is used as the weight of the feature, which is defined in Eq. (1).

$$TF - IDF : w_{ij} = (\frac{freq_{ij}}{MaxFreq_{ij}}) \times \log \frac{N}{n_i} \quad (1)$$

Here, w_{ij} is the weight of term (or phrase) t_i in document d_j, $freq_{ij}$ is the frequency of t_i in d_j, $MaxFreq_{ij}$ is the frequency of the term (or phrase) whose frequency is the most in d_j, N is the number of documents under one given name, and n_i is the number of documents which has term (or phrase) t_i.

In this study, the similarities based on features described above were calculated using K-L divergence defined as Eq. (2).

$$D_{KL}(P \parallel Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \quad (2)$$

P and Q denote the vector of a document respectively. K-L divergence between two vectors shows the distance of two related documents. The smaller the value of K-L divergence of two vectors becomes, the closer the two documents are. In order to prevent the zero denominator, we applied Dirichlet smoothing, i.e. , the zero element in the vector will be replaced by 0.00001.

2.3 Clustering

Clustering is the key part for our personal name disambiguation system. This task is viewed as an unsupervised hard clustering problem. First, we view the problem as unsupervised, using the distributed training data for parameter validation, to optimally tune the parameters in the clustering algorithm. Secondly, we observed that the majority of the input documents reference a single individual. Hence, we view the problem as hard clustering, assigning input document to exactly one individual, so that the produced clusters do not overlap.

In our system, hierarchical agglomerative clustering (HAC) is used as a clustering method. It builds up a hierarchy of groups by continuously merging the two most similar groups. Each of these groups starts as a single item, in this case an individual document. In each iteration this method calculates the distances between every pair of groups, and the closest ones are merged together to form a new group. The vector of the new group is the average of the original pair. This is repeated until there is only one group. This process is shown in Fig. 2.

We used a threshold for selecting cluster. So it is not necessary to determine the number of clusters beforehand. We view the whole group as a binary tree, every node which is not a leaf has two children, left child and right child, and has a record of the distance between the two children. We traverse the tree from the root, if the distance between the pair of children which form the cluster is larger than the threshold, then move down to check the distance of its left child, then right child. The process will continue until the distance between two children is less than the threshold. When the process comes to an end, all the leaves under the node will be considered to be in the same cluster. The selecting process will continue until all the leaves are assigned to a cluster. The threshold is tuned using the distributed training data.

The whole process mainly consists of two phases, the first phase is clustering all the single items into one group, and the second is selecting cluster down along the tree from the root. This strategy has a major disadvantage which is the new node is the average of its children. Hence, with the merger of nodes going on, the distance between different groups becomes smaller and smaller, which makes the boundaries between different clusters blur. This is probably the main reason that leads to the unsatisfactory results.

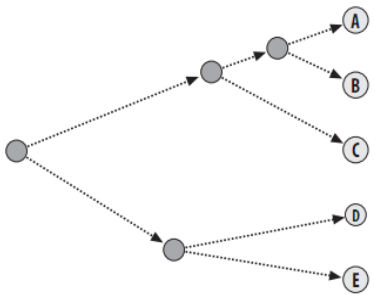

Figure 2 visualization of hierarchical clustering

3 Performance

Since there is no correct answer of test data received, we present the performance of our system of training data. There are two results gotten from the distributed evaluation in Table 1: one is evaluated with B-Cubed, and the other with P_IP. Both scores indicate that personal name disambiguation needs more effort.

Table 1 The performance of training data

	precision	recall	F_score
B-Cubed	71.83	62.88	56.98
	purity	Inverse purity	F_score
P_IP	76.43	67.71	62.76

4 Conclusion

In this report, we describe a system for the Chinese Personal Name Disambiguation task, applying a two-stage clustering model. Because this is our first time attending this kind of task, there are many aspects not having been taken into account. Therefore, improving system performance becomes motivation for us to work on it continuously. In future work, we'll focus on improving the clustering algorithm and proper feature extraction.

References

J. Artiles, J. Gonzalo and S. Sekine. WePS 2 Evaluation Campaign: overview of the Web People Search Clustering Task. In 2nd Web People Search Evaluation Workshop (WePS 2009). In18th WWW Conference, 2009.

http://ir.hit.edu.cn/

A. Bagga and B. Baldwin. 1998. Entity–based Cross–document Co–referencing Using the Vector Space Model. In 17th COLING.

C. H. Gooi and J. Allan. 2004. Cross-Document Coreference on a Large Scale Corpus.NAACL

T. Pedersen, A. Purandare and A. Kulkarni. 2005. Name Discrimination by Clustering Similar Contexts. In Proc. of the Sixth International Conference on Intelligent Text Processing and Computational Linguistics, page 226-237. Mexico City, Mexico.

Y. Chen and J. H. Martin. CU-COMSEM: Exploring Rich Features for Unsupervised Web Personal Name Disambiguation. In WWW Conference, 2007.

M. Ikeda, S. Ono, I. Sato, M. Yoshida and H. Nakagawa. Person Name Disambiguation on the Web by TwoStage Clustering. In 18th WWW Conference, 2009.

E. Elmacioglu, Y. F. Tan, S.Yan, M. Y. Kan and D. W. Lee. Web People Name Disambiguation by Simple Clustering with Rich Features. In WWW Conference, 2007.