Web Personal Name Disambiguation Based on Reference Entity Tables Mined from the Web

Xianpei Han  Jun Zhao
Institute of Automation, Chinese Academy of Sciences
Haidian District, Beijing, China
+86 10 8261 4468
{xphan, jzhao}@nlpr.ia.ac.cn

ABSTRACT
Ambiguous personal names are common on the Web, which pose a challenge for many different tasks. The traditional disambiguation employs the clustering methods. However, without reference entity tables, the clustering method can only identify whether two names refer to the same entity, rather than identify which entities they refer to. Furthermore, clustering methods are difficult to achieve robust performance on different names. Some recent disambiguation methods (the link-with-entity-base methods) extract the reference entity tables from online entity bases. The link-with-entity-base methods, however, suffer from the entity base’s limited coverage problem, so it can only disambiguate names in a limited coverage.

In this paper, to overcome the previous methods’ deficiencies, we propose a web-querying method to mine the reference entity tables from the Web automatically with the help of professional category knowledge. Then, we disambiguate personal names by linking them to the personal entities within the mined tables through categorization. The experimental results on the dataset extracted from Freebase show that our web-querying method can effectively mine personal entity with an F-measure 0.90. The disambiguation results on WePS datasets show that our method can achieve more robust and informative performance than the traditional clustering methods; and outperforms the traditional link-with-entity-base methods with a 0.29 improvement in F-measure.

Categories and Subject Descriptors
H.3.3 [Information Systems]: Information storage and retrieval—Information Search and Retrieval.

General Terms
Algorithms, Experimentation.

Keywords
Named Entity Disambiguation, Name Disambiguation, Person Resolution, Web Person Search.

1. INTRODUCTION
Ambiguous personal names are common on the Web. According to Wan et al.[20], 68% names are ambiguous in the 200 most frequent person queries of MSN. For example, the name “Michael Jordan” refers to more than ten persons in the Google search results. Some of them are shown below:

Michael (Jeffrey) Jordan, Basketball Player
Michael Jordan, Footballer
Michael (J.) Jordan, Professor of Berkeley

The ambiguous names pose a challenge for many different tasks. For example, in response to a person query, search engine returns a long, flat list of results containing web pages about several namesakes. The users are then forced either to refine their query by adding terms, or to browse through the search results to find the information he is looking for. Besides, an ever-increasing number of question answering, information extraction systems are coming to rely on the web data, ambiguous names will lead to wrong answers and poor results. Therefore, name disambiguation is critical in helping users and applications to find the information they are looking for.

In order to disambiguate personal names, a method must identify which personal entities the personal names refer to. So the reference entity table, which contains all the possible reference entities of a particular name (for example, a table contains all persons named Michael Jordan), plays an important role in the disambiguation process. Unfortunately, the reference entity tables are usually unknown or incomplete. So the traditional name disambiguation employs clustering methods, which simply cluster all the appearances of a particular name, with each resulting cluster corresponding to one person [Bagga and Baldwin[1]; Mann and Yarowsky[9]; Fleischman[14]; Pedersen et al.[19]].

However, without reference entity tables, clustering methods can only identify whether two names refer to the same entity, rather than identify which entity they refer to. Furthermore, clustering methods need to set some sensitive parameters, such as the end condition (the cluster number, the similarity threshold, etc.). Cluster methods usually train a parameter setting on several names and apply it to disambiguate other names, which usually cannot achieve robust disambiguation performance because of the diversity of names.

In order to extract the reference entity tables, several recent disambiguation methods (the link-with-entity-base methods) have employed the online entity bases such as Wikipedia and DBLP. In particular, they first extract the reference entity tables from large online entity bases such as Wikipedia and DBLP. Then the ambiguous names are disambiguated by linking them to particular
entities in the tables. (Bunescu and Pasca[15]; Cucerzan[17]; Hassell et al.[12]). The link-with-entity-base methods, however, suffer from the entity base’s limited entity coverage problem: there is no entity base which can cover most of the persons on the web; besides, most entity bases are only built for a special domain, such as the DBLP for research domain. So the link-with-entity-base method can only disambiguate names in a limited coverage.

In this paper, we propose a novel personal name disambiguation method, which can automatically mine reference entity tables from the Web. The starting point of our method is that professional categories can be used to distinguish the reference entities of a name. For example, we can distinguish the reference entities of Michael Jordan as Michael Jordan (Basketball player), Michael Jordan (Politician), etc. Starting from this idea, we propose a novel web-querying method to mine reference entity tables from the Web. Using the mined tables, personal names are disambiguated by linking them to the entities in the mined tables through categorization.

Compared with the traditional methods, our method has the following advantages:

1) Compared with the clustering methods, our method has two advantages. First, based on the reference entity tables mined from the Web, we can identify the reference entities of the names to be disambiguated. Second, our method can achieve more robust performance, which is important in real-world applications. The reason lies in the fact that the reference entity tables are specialized for different names, so they can capture the diversity of different names. We believe both these advantages can greatly improve real-world applications’ performance, for example, in web person search, the web page clusters with target entity description can greatly improve the efficiency of users’ web page choosing, and the robust performance makes the disambiguation method can be applied in real-world applications.

2) Compared with the link-with-entity-base methods, our method is full coverage of personal names. For every name, the reference entity table is mined from the Web automatically, it doesn’t depend on a given entity base.

There were also several other research which explores the usage of the Web for disambiguation. For example, Kalashnikov[6] enhanced the similarity measure between two name appearances by collecting their web co-occurrence information; Tan et al.[18] mined contextual information for linking short forms to longs; Tan et al.[18] represented the citation with retrieved URLs in author disambiguation. However, all the above research work only enhances the similarity measures for clustering. In contrast, our method differs from them by focusing on mining reference entity tables.

To mine the reference entity table of a particular name, our method needs to search the Web $N$ times, where $N$ is the size of professional category. This is a timing consuming operation, unless it is performed at a server (Web Search Engine) side (Kalashnikov et al.[6]). So for applications with real-time demand, it is more feasible and effective to implement our method in server-side. In addition, as more and more search engines open their search API with fewer restrictions, such as the Yahoo! Search BOSS API (with unlimited query per day), it is also feasible to implement our method in client-side. Our method can also be combined with the link-with-entity-base method as they are good complementary of each other.

This paper proceeds as follows. Section 2 reviews the related work. Section 3 describes our proposed method. Section 4 includes the experimental results and discussions. This paper ends with a review of summary and future directions.

2. Related Work

In this section, we briefly review the two traditional methods used in web personal name disambiguation and named entity disambiguation, which is highly related to our work.

2.1 The Clustering Methods

The clustering method group a particular name’s all appearances into different clusters, with each resulting cluster corresponds to one person. In particular, the clustering methods first compute the similarities between different name appearances, then cluster them using a predefined end condition.

One clustering method disambiguates names based on their context similarity. Bagga and Baldwin[1] represents a name as a vector of its context words, the similarity between two names was determined by the co-occurring words, then two names were predicted to be the same entity if their similarity scores are above a threshold. Mann and Yarowsky[9] extended the name’s vector presentation by extracting the structured biographic facts. Ted Pedersen et al.[19] employed significant bigrams to represent the contexts of a name. Fleischman[14] trained a Maximum Entropy model to give the probability that two names refer to the same entity, then use a modified agglomerative clustering algorithm to cluster names using the probability as the similarity.

Another clustering method was based on the similarity computed using the relation information or the link structure in a social network. Bekkerman and McCallum[16] disambiguated names based on link structure of the Web, their model leverages hyperlinks and the content distance between pages. Malin and Airdoll[3] and Malin[2] measured the similarity based on the probability of walking from one ambiguous name to another in the social network constructed from all documents. Minkov et al.[7] disambiguated names in email documents by building a graph from the email data, then employs a lazy graph walk to compute the similarity.

The clustering methods are easy-to-use, but it can only identify whether two names refer to the same entity, rather than find the reference entity of a name. The previous researches using clustering methods focus on choosing a better similarity measure. However, a remaining question is how to determine the optimal parameter setting, especially in an open environment such as the Web (Chen and Martin[22]). Another problem is that a special step is needed to produce an informative description for each cluster (Wan et al.[20]).

2.2 The Link-with-Entity-Base Methods

The link-with-entity-base method linked ambiguous names with unambiguous entities in a given entity base. Bunescu and Pasca[15] disambiguated the names in Wikipedia by linking them to the most similar Wikipedia entity using the similarity computed using a disambiguation SVM kernel. Cucerzan[17] disambiguated names through linking them to Wikipedia entities by comparing
their vectorial representations. Joseph et al.[12] used various relationships in a document as well as from a given ontology to pinpoint names to the persons in the DBLP.

The link-with-entity-base method works well when the entity base covers most of the reference entities, but this restriction is usually too strict.

3. Our Method
Given a personal name, the disambiguation tasks are: 1) mining the reference entity table \(P=\{p_1, p_2, ..., p_n\}\) of the name; 2) linking the appearances of the name to the entities in the table. This paper makes the standard “one person per document” assumption, i.e. all the appearances of a name in a document are assumed to refer to the same entity, so the second task is simplified as linking web pages to personal entities.

Figure 1. The proposed method’s framework

In order to mine reference entity tables from the Web, we first propose a novel web-querying method. Then web pages are linked to personal entities through categorization. Figure 1 plots the method’s framework with an example of disambiguating Michael Jordan. In the following, we first demonstrate how to employ professional category for name disambiguation, then describe our person disambiguation method in detail.

3.1 Employing Professional Category for Name Disambiguation
In this paper, we represent a personal entity by his name combined with professional category information. For example, we represent the different reference entities of Michael Jordan as Michael Jordan (Basketball player), Michael Jordan (Footballer), and Michael Jordan (Politician), etc. This is also the disambiguation and representation strategy used in Wikipedia. The professional categories can be obtained in many ways. For example, we can use the Standard Occupational Classification system; or extract it from online knowledge bases, such as Freebase and Wikipedia.

In this paper, we use a professional taxonomy extracted from the Freebase\(^1\), which is a structured knowledge base contains more than 800 thousands of personal entities labeled with more than 2,000 professional categories. In particular, the extraction process is as follows. First, all professional categories\(^2\) in Freebase are extracted, then the professional categories which are too general (such as “person”) or too specific (such as “Mayor of Wellington”) are manually eliminated. Totally, 1,712 professional categories are extracted.

![Image](http://www.freebase.com/view/people/views/profession)

Table 1. Statistics of the Freebase People Base

We demonstrate the disambiguation capacity of the professional category using some statistics of the Freebase People Base\(^3\) shown in Table 1. As seen in Table 1, we can see that:

1) The professional category has a good disambiguation capacity. Only 914 (out of the 40,074, fewer than 3%) ambiguous names’ reference entities cannot be distinguished using professional categories, that is, some different persons with the same name may have a same profession.

2) Most personal entities can be represented by its name with single professional category. In the set of the personal entities attached with professional categories, nearly 80% have single professional category, while only about 20% have multi-professional categories.

3.2 Mining Reference Entity Tables using a Web-Querying Method
As the first task of disambiguation, we need to mine the reference entity tables of different personal names. To solve this task, we propose a web-querying method. We consider two types of reference entities of a personal name:

The single-profession person. The personal entity can only be categorized into a single professional category. For example, the English footballer Michael Jordan has a single professional category footballer. As shown in Table 1, the single-profession person accounts for about 80% personal entities in Freebase.

The multi-profession person. The personal entity can be categorized into several different professional categories. For example, the Pop singer Michael Jackson can be categorized into singer, songwriter and dancer categories. The multi-profession person accounts for about 20% personal entities in Freebase.

In order to mine all the single-profession persons of a personal name, we first list all candidates by combining the name with each professional category. For example, all the Michael Jordan’s single-profession person candidates are listed as {Michael Jordan (Basketball player), Michael Jordan (Footballer), Michael Jordan (Physicist), …}. Then we verify the existence of these candidates through collecting evidence from web.

Based on single-profession persons mined from the Web, we extract the multi-profession persons by merging the single-profession persons who are actually the same person through collecting evidences.

\(^1\) http://www.freebase.com
\(^2\) http://www.freebase.com/view/people/views/profession
\(^3\) 1 http://www.freebase.com/view/people/views/person
3.2.1 Single-profession Person Mining

Given all the single-profession person candidates of a personal name, we collect the existence evidence of them through web-querying. Then we verify them using these collected evidences.

Web Evidence Collecting through Web-Querying. On the Web, the existence evidence of a single-profession person is the web page which mentions him, which we called evidence page. The data on the Web is redundant, so we can find the most obvious evidence pages using the follows intuitive definition:

A web page is an evidence page of a single-profession person if in this web page there is at least one sentence which contains both the name and the professional category name.

For example, the sentence “Michael Jordan is the greatest basketball player of all time” within the Wikipedia Michael Jordan page indicates this page is an evidence page of Michael Jordan (basketball player).

We collect the evidence pages of a single-profession person from the web through querying. We illustrate this process using an example, which finds evidence pages of Michael Jordan (basketball player):

1) We build a heuristic search query using the name together with the professional name, i.e. “Michael Jordan” + “basketball player”.

2) The query is submitted to a search engine (Yahoo! in this paper) 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, as shown in Figure 2, are confirmed to be evidence pages.

Figure 2. The Top 3 Yahoo! search results of “Michael Jordan” + “basketball player”

So, using the top $N$(which is learned during a supervised process) search results, we can collect a set of evidence pages for every single-profession person candidate.

Verification. The more evidence pages indicate the higher existence probability of a single-profession person. So we use an evidence page count threshold to filter out these single-profession person candidates with few or no evidence pages. The threshold value is learned during a supervised learning process.

3.2.2 Multi-profession Person Mining

The above Single-profession Person Mining process extracts a multi-profession person as several separate single-profession persons, for example, extracts the pop singer Michael Jackson as Michael Jackson(singer), Michael Jackson(Songwriter) and Michael Jackson(dancer). So we can obtain multi-profession persons through merging single-professional persons. Below, we describe the details.

The Merging Evidence. To merge two persons $p_i$ and $p_j$, we need to identify whether they are the same one. We use the Jaccard similarity between their evidence page set, i.e., $p_iEP$ and $p_jEP$ as the merging evidence:

$$\text{Similarity}(p_i, p_j) = \frac{|p_iEP \cap p_jEP|}{|p_iEP \cup p_jEP|}$$

Merging. Two persons $p_i$ and $p_j$ are merged if the merging evidence of them is larger than a given threshold(the value is learned during a supervised learning process). The resulting multi-profession person is represented by its name combined with all professional categories of the merged personal entities, and its evidence page set is the union of the evidence page sets of the merged personal entities.

3.3 Disambiguation by Categorization

Based on the mined reference entity tables, we disambiguate a name’s appearances (web pages here) by linking them with the personal entities within its reference entity table. We regard the linking process as a categorization task: 1) To build the training set for every mined personal entities; 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.

Building the training set. For every personal entity in the reference entity table, we use its evidence page set as its training set.

Extracting the feature representation. A web page is represented as a vector of features as follows.

Tokens. Identical with the systems in Javier Artiles et al.[10], we segment the web page’s text content into words and then stem them using the Porter stemmer, stop words are filtered. Each word retained is used as a feature and weighted by its Term Frequency × Inverse Document Frequency (TF×IDF).

Named Entities. We extract the named entities from the web page using the OpenNLP(http://opennl.sourceforge.net/) Named Entity Detection tools. 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 such as the topic of the web page. 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.

Categorization via kNN Classifier. Given the reference entity table $(p_1, p_2, \ldots, p_n)$, a web page $wp$ is linked to a personal entity $p$ using the kNN classifier (Dasarathy[5]) according to the formula:

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

where the similarity between a personal entity and a web page is determined by the max cosine similarity between the web pages in the training set of the personal entity and the web page to be classified.
4. Experimental Results and Discussions
To assess the performance of our method and compare it with traditional methods, we conduct a series of experiments. In the following, we first evaluate the reference entity table mining performance of our web-querying method in Section 4.1, then evaluate our method’s name disambiguation performance and compare it with traditional methods and the state-of-art performance in Section 4.2. The analysis of the experimental results is also given.

4.1 The Results of Reference Entity Table Mining
The name disambiguation method is depending on the reference entity tables mined from web. In this section, we evaluate the performance of our web-querying method for reference entity table mining: we first describe the used data sets and the web querying method’s configuration. Then we evaluate the web querying method’s performance. We also show several names reference entity tables and compare it with Freebase data.

4.1.1 Data sets and Configuration
As there is no data provides the complete reference entity tables of names, we evaluate the reference entity table mining performance of our web-querying method by how well it mined personal entities. We use several data sets randomly extracted from Freebase as follows:

**EP:** A set of 100 existent one-profession persons. This data set will be used to evaluate the recall of the web-querying mining method;

**NEP:** A set of 100 forged one-profession persons, all persons are non-existent. Together with **EP,** this data set will be used to evaluate the precision of the web-querying mining method. We forge a nonexistent one-profession person through randomly selecting a personal name and a professional category from the Freebase, then manually verify its nonexistence on the web.

**CP:** A set of 100 single-profession person pairs, within each pair the two persons is the same one but with different professional categories, merge such a pair will result in a right multi-profession person. This data set will be used to evaluate the recall of multi-profession person.

**NCP:** A set of 100 single-profession person pairs, within each pair, the two persons sharing the same name but are different one, merge such a pair will result in an inaccurate multi-profession person. This data set together with the **CP** data set will be used to evaluate the precision of the mined multi-profession person.

For all the above data sets, we use 50% for configuration, and the other 50% for testing.

**Configuration.** Configuring the web-querying method involves setting three parameters:

The first parameter specifies the top N search results (Freebase pages and Wikipedia pages are filtered) which are considered for finding the web evidences. More search results provide more information, but needs more search time. Figure 3 plots the

\[
\text{Similarity(p, wp)} = \max_{ep \in p, EP} \text{Cosine}(ep, wp)
\]

tradeoff (we choose the N according to how many search counts are needed for obtaining it). We identified the top 20 results contains enough information for reference entity table mining.

The second parameter specifies the threshold of evidence page count for single-profession person verification. A larger threshold will filter out more nonexistent entities but recall fewer existent entities. Figure 4 plots the tradeoff. We identified 1 as a sensible threshold where the F-measure achieves the best.

The third parameter sets the merging threshold for multi-profession person mining. A larger threshold will result in higher merging precision but lower merging recall. Figure 5 plots the tradeoff. As a wrong merging will result in more errors than a
missing merging, we identified 0.055 as a sensible threshold for it achieves both high F-measure and precision.

4.1.2 Results and Examples
In this section, we show our web-querying method’s reference entity table mining performance in Table 2. Several names’ mined reference entity tables are shown and compared with their reference entities in the Freebase in Table 3.

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>Precision</th>
<th>Recall</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-profession</td>
<td>0.98</td>
<td>0.84</td>
<td>0.90</td>
</tr>
<tr>
<td>Multi-profession</td>
<td>0.86</td>
<td>0.62</td>
<td>0.72</td>
</tr>
</tbody>
</table>

Table 2. The reference entity table mining performance

<table>
<thead>
<tr>
<th>Personal name</th>
<th>Mined Personal Reference entity Table</th>
<th>Personal reference entities in Freebase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abby Watkins</td>
<td>(Climbing, Athlete)</td>
<td>None</td>
</tr>
<tr>
<td>Andrew Powell</td>
<td>(Composer, Musician, Author, Writer)</td>
<td>(Musical Artist)</td>
</tr>
<tr>
<td></td>
<td>(President)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Professor, Economist)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Psychiatrist)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Journalist, Columnist)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>David Lodge</td>
<td>(Author, Writer, Literary Criticism)</td>
<td>(Novelist)</td>
</tr>
<tr>
<td></td>
<td>(Actor)</td>
<td>(Board Member)</td>
</tr>
<tr>
<td></td>
<td>(Voice actor)</td>
<td>(Film actor)</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Several names’ mined reference entity tables and their reference entities in Freebase

From Table 2 and Table 3, we can see that,

1) Our web-querying method can effectively mine the single-profession person with an F-measure 0.90 and the multi-profession person with an F-measure 0.72.

2) The mined personal entities are very accurate, with a precision of 0.98 for one-profession person and 0.86 for multi-profession person.

3) The Freebase misses many personal reference entities, for example, the climbing athlete Abby Watkins, the Ecologist David Lodge and the Journalist Andrew Powell are missed.

4) Our web-querying method can extract the missing personal entities and also the missing professional categories in the Freebase. For example, our method extracted the missed profession literary criticism for the novelist David Lodge.

4.1.3 Discussions
Based on the experimental results in Table 2 and 3, we can make the following observations:

1) The existing entity bases are unsuitable for web personal name disambiguation for their limited entity coverage. As seen in Table 3, lots of entities are not covered in the Freebase.

2) Our web-querying method can effectively mine the reference entity tables by extracting both the single-profession person and the multi-profession person with a high F-measure.

3) The high personal entity mining precision indicates it can be used in real applications.

4) Compared with the link-with-entity-base method, our method is more suitable for web personal name disambiguation task.

4.2 The Performance of Web Personal Name Disambiguation
The web personal name disambiguation method outputs a category for each appearance(web page here) of a personal name (for clustering methods, the category is the cluster it belongs to, for our method, the category is the reference entity) and all the web pages sharing the same category are assumed to refer to the same personal entity.

We adopt the measures in the First Web People Search Clustering Task (WePS1) (Javier Artiles et al.[8]) to evaluate the disambiguation performance. These measures are: the purity (Pur) which measures the homogeneity of the name appearances in the same category; the inverse purity (Inv_Pur) which measures the completeness of a category; the F-Measure (F) which is the harmonic mean of the purity and the inverse purity. The detailed definitions of these measures can be found in (Enrique Amigo et al. 2008). Within the three measures, the F-measure is the most important measure.

4.2.1 Data sets and Baselines
We adopt the data sets used in the First Web People Search Clustering Task (WePS1) and the Second Web People Search Clustering Task (WePS2). We use three data sets: the WePS1_training data set, the WePS1_test data set, and the WePS2_test data set. Each of the three data sets consists of a set of ambiguous personal names (totally 109 names), and every name’s appearances within the web pages of the top N (100 for WePS1 and 150 for WePS2) search results are needed to be disambiguated.

In order to compare our method with the traditional methods, we implement three baselines:

Clustering_Simi: The web pages are clustered using an agglomerative clustering algorithm over traditional vector space models (Bagga and Baldwin[1]) with a similarity threshold as the end condition. The web page is represented using the method described in Section 3.2.2. For every data set, we train the end condition using the other two data sets, and the better result is chosen as the final result. This baseline is also the state-of-art methods used in the WePS.

Clustering_K: The same as the Clustering_Simi except that the cluster number is used as the end condition.

Link_with_Freebase: The web pages are disambiguated by linking them with the personal entities in the Freebase using the
kNN classifier described in Section 3.2.3. The personal entities in Freebase are represented using its detailed page in Freebase and its Wikipedia page.

4.2.2 Our Method vs. The Clustering Methods
We compared our method with the two clustering methods in all three data sets. The results are shown in Table 4.

<table>
<thead>
<tr>
<th>Method</th>
<th>WePS1_training</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pur</td>
<td>Inv_Pur</td>
<td>F</td>
</tr>
<tr>
<td>Clustering_Simi</td>
<td>0.67</td>
<td>0.88</td>
<td>0.72</td>
</tr>
<tr>
<td>Clustering_K</td>
<td>0.90</td>
<td>0.61</td>
<td>0.67</td>
</tr>
<tr>
<td>Our Method</td>
<td>0.81</td>
<td>0.78</td>
<td>0.76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>WePS1_test</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pur</td>
<td>Inv_Pur</td>
<td>F</td>
</tr>
<tr>
<td>Clustering_Simi</td>
<td>0.64</td>
<td>0.93</td>
<td>0.74</td>
</tr>
<tr>
<td>Clustering_K</td>
<td>0.71</td>
<td>0.85</td>
<td>0.74</td>
</tr>
<tr>
<td>Our Method</td>
<td>0.74</td>
<td>0.84</td>
<td>0.75</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>WePS2_test</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pur</td>
<td>Inv_Pur</td>
<td>F</td>
</tr>
<tr>
<td>Clustering_Simi</td>
<td>0.61</td>
<td>0.79</td>
<td>0.66</td>
</tr>
<tr>
<td>Clustering_K</td>
<td>0.62</td>
<td>0.72</td>
<td>0.62</td>
</tr>
<tr>
<td>Our Method</td>
<td>0.72</td>
<td>0.83</td>
<td>0.73</td>
</tr>
</tbody>
</table>

Table 4. Our Method vs. the Clustering methods

From the Table 4, we can see that:

1) The clustering methods are difficult to achieve robust performance on different data sets. As shown in Table 4, the performance variations of the two clustering baselines are significant in different data sets.

2) Compared with the clustering methods, our method achieves more robust performance. As shown in Table 4, the performance variation of our method is slight. We believe this is because the mined reference entity tables are specialized to different names, as shown in Table 3.

4.2.3 Our Method vs. The Link-with-entity-base Method
We compare our method with the Link-with-Freebase method on a data set selected from WePS data, which only contains the ambiguous names contained in FreeBase (40 out of 109).

<table>
<thead>
<tr>
<th>Method</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pur</td>
<td>Inv_Pur</td>
</tr>
<tr>
<td>Link_with_FreeBase</td>
<td>0.44</td>
<td>0.94</td>
</tr>
<tr>
<td>Our Method</td>
<td>0.77</td>
<td>0.75</td>
</tr>
</tbody>
</table>

Table 5. Our Method vs. The link-with-entity-base method

From these results, we can see that:

1) The Link-with-entity-base methods are difficult to achieve good results on the open web because its coverage problem, as shown in Table 3 and 5, the missing of personal entities in the entity base leads to poor disambiguation results.

2) As shown in Table 5, our method achieves appealing results compared with the link-with-Freebase method, with a 0.29 improvement in F-measure.

3) Compared with the link-with-entity-base method, our method is more suitable for disambiguating personal names on the open web. First, our method is full coverage of personal names. Second, even for the names whose reference entities can be found in the given entity base, our method still outperforms it with a 0.29 improvement in F-measure, as shown in Table 5.

4.2.4 A Comparison with the State-of-art Systems
In Table 6 and Figure 6, we compared our method with the results of the systems participated in WePS1. In comparison with the total 16 systems, the performance of our method can achieve the second-best performance. Till now, our method only uses the features which are rather straightforward. We believe our method can be improved further by using more rich features and better feature weight algorithms.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Team-id</th>
<th>Pur</th>
<th>Inv_Pur</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CU_COMSEM</td>
<td>0.72</td>
<td>0.88</td>
<td>0.78</td>
</tr>
<tr>
<td>2</td>
<td>Our Method</td>
<td>0.74</td>
<td>0.84</td>
<td>0.75</td>
</tr>
<tr>
<td>3</td>
<td>PSNUS</td>
<td>0.73</td>
<td>0.82</td>
<td>0.75</td>
</tr>
<tr>
<td>4</td>
<td>UVA</td>
<td>0.81</td>
<td>0.60</td>
<td>0.67</td>
</tr>
<tr>
<td>5</td>
<td>SHEF</td>
<td>0.60</td>
<td>0.82</td>
<td>0.66</td>
</tr>
</tbody>
</table>

Table 6. A comparison with WePS1 results

As shown in Figure 6, the performance of WePS1 systems varies in a wide range, from 0.40 to 0.78. As all systems participated in WePS1 employ clustering methods, we believe the wide performance variation reflects the difficult of choosing the optimal parameter setting for clustering methods.

5. Conclusion and Future Work
This paper is the first attempt to disambiguate personal names by mining the reference entity tables from the Web. Based on the mined reference entity tables, our method can disambiguate names by identifying their reference entities, achieve more robust performance and is full coverage of names. Furthermore, the
The proposed method only involves human efforts in choosing professional category knowledge.

On the WePS data sets, our method achieves appealing performance for personal name disambiguation:

1) Compared with the clustering methods, our method can identify which entity a name refers to, and achieve more robust performance. We believe both these improvements can greatly improve real-world applications’ performance, for example, in web person search, the web page clusters with target entity description can greatly improve the efficiency of users’ web page choosing, and the robust performance makes the disambiguation method can be applied in real-world applications.

2) Compared with the link-with-entity-base methods, our method is full coverage of names and can outperform the link-with-entity-base methods with a 0.29 improvement in F-measure. The disambiguation method used in this paper can also be used in disambiguating entities of other types (e.g., location, organization) by selecting an appropriate taxonomy. Furthermore, we believe that the mined reference entity tables will also be useful in many other different tasks, such as knowledge base population, ontology population with instances.

6. ACKNOWLEDGMENTS

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

7. REFERENCES