PERSONAL NAME DISAMBIGUATION IN FARSI WEB PAGES

Hojjat EMAMI[1]

The problem of name ambiguity causes the results for a personal name query to be a mixture of web pages about different individuals sharing the same name. Name disambiguation as an important task of web mining and information retrieval is the process of grouping web pages into some clusters, where each cluster contains all web pages that refer to the same individual. This paper presents an unsupervised approach to name disambiguation problem. The proposed method exploits two sources of semantic information: discourse profile information derived from the local corpus and global information extracted from ontology. Our approach formalizes the name disambiguation problem as four main subtasks: pre-processing, discourse profile extraction, profile enrichment and profile clustering. First, our approach takes as input the web pages to be disambiguated and then cast them into annotated textual documents using pre-processing tools. Profile extraction phase extracts individuals' discourse profiles from pre-processed text. In profile enrichment phase, discourse profiles are enriched with semantic information obtained from an online ontology. In profile clustering stage, enriched profiles are grouped into some clusters such that each cluster contains the web pages refer to the same individual. The performance of the proposed approach is evaluated on a Farsi and English datasets. The experimental results are encouraging and indicate that the proposed method outperforms the baseline methods and its counterparts.

Keywords: web mining, information retrieval, name ambiguity, name disambiguation, Farsi language, clustering

1. Introduction

In recent years, the number of websites and social media has increased significantly because of the increased Internet penetration and the increased number of Internet users [1]. As a result, a vast amount of valuable data about various entities is generated on the web and various social media. These data are in various languages and contain information about persons, organizations, locations, governments and many other entities. Users to reach their desired information on the web usually use different search engines. Searching for personal names is among the most frequently queried items in search engines. The results retrieved from a search engine for a personal name are a set of web pages, in ranked order, where each page is assumed relevant to the query name.

[1] University of Bonab, Islamic Republic of Iran, e-mail: emami@bonabu.ac.ir

As personal names are highly ambiguous, personal information retrieval systems deal with a fundamental problem, namely *name ambiguity* problem [2]. The problem of name ambiguity causes the results of a personal name search to be a mix of web pages about different people sharing the same name. For example, the search results for name "احمد عبدالهزاده/ Ahmad Abdollahzadeh" are a set of web pages relevant to any person with name "احمد عبدالهزاده". This name in the first case refers to a famous football defensive midfielder (https://fa.wikipedia.org/wiki/احمد_عبدالله‌زاده) and in the second case refers to an academic professor (http://ceit.aut.ac.ir/~ahmad/).

Name disambiguation systems have emerged in response to the name ambiguity problem. Let $W = \{W_1, W_2, ..., W_N\}$ to be a collection of N web pages in the search results for a person name, and $M = \{m_{11}, m_{12}, ..., m_{21}, m_{22}, ...\}$, $m_{ij} \in W_i$ to be a set of name observations within document collection W, which need to be disambiguated. The goal of name disambiguation is to group web pages in collection W into K disjoint clusters $G = \{C_1, C_2, ..., C_K\}$, where $\forall\, i,\ C_i \neq \varnothing$, $\forall\, i \neq j\ \ C_i \cap C_j = \varnothing$, and $\bigcup_{i=1}^{K} C_i = W$. The web pages in each cluster $C_i \in G$ only refer to the same individual in reality.

There are many ongoing researches on the problem of name disambiguation in different languages. However, one of the less studied languages in name disambiguation is Farsi. The volume of Farsi digital content on the web has increased at a steady rate over the past years and constituents 1% of the current web content [3]. Processing Farsi content is more difficult because of the special characteristics of Farsi language and the lack of natural language processing tools and linguistic resources. These challenges imply the necessity of developing powerful name disambiguation systems in Farsi. In this article, we present a semantic approach to name disambiguation in Farsi and report an evaluation of that.

In recent years, a few efforts have been made to automatically cluster Farsi web pages [4–7]. These approaches did not completely discuss name disambiguation problem and often focus on web page clustering. Recently, several works have been conducted to automatically disambiguate personal names in English context. The majority of these approaches use a combination of various features to compute similarities between web pages and then use various clustering algorithms to disambiguate names. However, these approaches suffer from several fundamental issues. The first is that the most existing name disambiguation approaches [8–13] exploit local syntactic and semantic features derived from the given local corpus. However, the local information may not be sufficient to resolve ambiguities and the robustness of system will be degraded due to the low quality of tools used to extract local features and incomplete information contained in local web pages. To alleviate these problems, some authors [10, 14, 15] besides the local information have utilized the global information derived from online corpora. Nevertheless, these solutions did not completely utilize all the semantic information contained in web pages such as entities' profile attributes. The second problem is that the existing work did not analysis semantically the information contained in local web pages. The third problem is that the results are surface clusters of persons that they are not linked to a proper ontology.

In this paper, we attempt to alleviate the deficiencies of previous work by proposing a personal name disambiguation approach in Farsi that not only uses discourse profile information obtained from the local corpus, but also semantic information about persons derived from an online ontology. Further, the named entities and concepts in the disambiguated text are linked to unique entries in an online ontology, which making it possible for software agents to easily understand, process and translate the text written in Farsi. The local and ontological profile attributes are two rich sources of information that can complement each other. This leads to more precise personal name disambiguation and confirms that a framework for enriching discourse profiles with online ontology information is needed.

To summarize, our contributions in this article lie in the approaches we propose to solve subtasks of name disambiguation problem:

- We present a semantic approach to personal name disambiguation problem that integrates the local personal information contained in given corpus and the information exist in BabelNet ontology [16]. Specifically, we present a profile enrichment method relying on deep semantic analysis of the textual content of web pages to deal with the problem of data sparseness and to make name disambiguation system more robust. Our approach is the first attempt to use an online multilingual ontology in the field of Farsi information retrieval.
- To evaluate the performance of the proposed approach, we build a small Farsi person name disambiguation dataset. We compare our approach with baseline and counterpart methods. The experimental results show that our approach is an encouraging approach, i.e., it can efficiently disambiguate names and cluster web pages with high quality.

The rest of this article is organized as follows. After a brief review of previous related work in Section 2, we describe our approach in Section 3. In Section 4, we describe the experiments performed to evaluate our approach. Finally, in Section 5, we draw some conclusions and identify future work.

2. Related work

Since there are a few name disambiguation approaches in Farsi, in the following, we first review some of the most relevant approaches in English and then discuss the existing works in Farsi.

The majority of recently proposed approaches in English often formulated name disambiguation as a clustering problem [9, 10, 14, 15, 17] Clustering methods have superior efficiency in dealing with a large volume of data and are useful when there is not a large labeled corpus. Clustering methods often includes three mains steps: feature extraction, similarity computation and object grouping. In most of the existing approaches, employed features are either syntactic or semantic. Syntactic features include tokens [8], specific keywords [9, 15], n-gram features, snippet-based features [10], etc. Semantic features include personal attributes [11, 12], hyperlinks [13], named-entities [9], etc. Each web page, which needs to be disambiguated, is represented as a vector of clustering features. Similarities among vectors are then computed using various similarity metrics to identify whether they refer to the same entity. Similarity computation forms the basis of name disambiguation in clustering approaches. Many existing approaches [11, 18–20] compute similarity between entities by matching their profiles. However, such methods ignore some important semantic information exist in external knowledge bases.

Some other approaches have harnessed social relations among entities to compute similarities [13, 21, 22]. These approaches often create a social relationship graph of entity names co-occurring in a document and then partition the graph into sets of clusters using graph clustering algorithms. The idea behind social network-based approaches comes from the fact that socially connected entities might be having the similar characteristics. The main limitation of these methods is that they may fail to disambiguate entities when a web page does not contain any information about people relationships.

Utilizing external features for name disambiguation was also studied in previous works [10, 14, 23–25]. Some of these methods [15, 25] exploit both the local information contained in given local corpus and the global information existing in online ontologies. Similar to [15], our approach exploits both the local information in the given local corpus and the global information in a online ontology for co-referent entities. Our approach relies on deep semantic analysis of the textual parts of web pages to extract ontological information, and the closeness centrality metric [17] in computing similarities among people attributes. There are also a few name disambiguation works in Farsi. Emami et al. [18] used two simple heuristics for cross-document name disambiguation, which include: (*i*) string matching of the personal names or aliases and (*ii*) matching entities' profile attributes. In first case, the similarity between any two personal names is computed using Edit distance and if the similarity is higher than a pre-defined threshold, the names are merged. In the second heuristic, similarity between entities is computed by matching their discourse profiles containing attributes of those entities. This heuristic is efficient; however, it ignores some important information contained in the online ontologies. Our approach extends this work through integrating the discourse profiles extracted from the given local corpus with the semantic information derived from BabelNet ontology.

3. Proposed method

Fig. 1 shows the main steps of our person name disambiguation approach. We formalize the name disambiguation problem as four subtasks: (*i*) pre-processing, (*ii*) profile extraction, (*iii*) profile enrichment, and (*iv*) profile clustering. In the following, we describe these components in more detail.

Procedure Name-Disambiguation (*e, W, O*)	
Input:	
E	▷ ambiguous name
$W = \{W_1, W_2, ..., W_N\}$	▷ input corpus
O	▷ online ontology
Output:	
$G = \{C_1, C_2, \ldots, C_K\}$	▷ disjoint name clusters
Initialization:	
$D \leftarrow \varnothing$	▷ pre-processed documents
$P \leftarrow \varnothing$	▷ discourse profiles
$P^+ \leftarrow \varnothing$	▷ enriched discourse profiles
$D \leftarrow$ Pre-processing (e, W)	
$P \leftarrow$ Profile-Extraction (e, D)	
$P^+ \leftarrow$ Profile-Enrichment (e, D, O, P)	
$G \leftarrow$ Profile-Clustering (e, P^+, Q)	▷ Q is similarity measure
Return *G*	

Fig. 1. Our proposed name disambiguation approach

Pre-processing

Let $E = \{e_1, e_2, ..., e_K\}$ be the set of named entities that need to be disambiguated, $W = \{W_1, W_2, ..., W_K\}$ be the input corpus, where each $W_i = \{w_1, w_2, ..., w_m\}$ be the set of web pages related to an ambiguous entity mention $e_i \in E$. In every iteration, pre-processing takes as input web page collection $W_i \in W$ related to entity mention $e_i \in E$ and prepared them according to system's desired format. The output of pre-processing step is a set of pre-processed and annotated textual documents $D = \{D_1, D_2, ..., D_K\}$, where

$D_j = \{d_1, d_2, ..., d_m\}$. Each $d_j \in D_i$ indicates the pre-processed document corresponding to web page $w_j \in W_i$.

In this article, we focus on the textual part of the Web pages, because the majority of the information about entities on the web is often expressed in the natural language text. Pre-processing consists of three main stages: (*i*) html tag removal, (*ii*) named entity recognition and (*iii*) co-reference resolution.

In html tag removal phase, for each web page $w_j \in W_i$, Jsoup[2] html parser is run to transform it to textual document $d_j \in D_i$. Named entity recognition phase takes as input documents and identifies coarse-grained lexical entity types using Polyglot-NER [27], a multi- lingual named entity tagger system. Common entity types include person, organization and location. For each named entity, we assign a unique index to distinguish the identity of entity. The annotated documents are passed to a rule-based co-reference resolution module to identify co-reference chains for the named entities mentioned in each documents. The mentions in every co-reference chain of interest are replaced with their corresponding representatives. Next, for the co-reference chain of interest within each document $d_j \in D_i$, we extract all sentences from the document d_j, create a summarized text by concatenating these sentences and then replaced the summarized text with document d_j.

The pre-processing tools may produce errors, which propagate to the later stages. However, improving the pre-processing components is beyond the scope of this article. The remainder of the processing described in the following sections uses this pre-processed text.

Profile extraction

Profile extraction function φ takes as input the entity mention e_i and its relevant pre-processed document $D_i = \{d_1, d_2, ..., d_m\}$. It then extracts values of different attributes from document $d_j \in D_i$ and forms a discourse profile P_j.

$$P_j \leftarrow \varphi(e_i, d_j) \tag{1}$$

P_j presents the discourse profile of the person entity e_i. P_j includes a number of <*a, v*> pairs, each of which represents a certain characteristic of entity e_i.

$$P_j = \{(a,v) | a \in A, v \in V(a)\} \tag{2}$$

a indicates an attribute of person e_i, *v* indicates a possible value for attribute *a*, *A* is the vocabulary of attributes that used to describe characteristics of the person e_i, and $V(a)$ presents a set of filler values for attribute $a \in A$.

[2] Jsoup: Java HTML Parser, [http://jsoup.org/]

For profile extraction, we used a semantic rule-based profile extraction method presented in [3]. This method is based on hand-crafted rules and uses two types of attribute extraction (AE) rules: verb-based AE (*VAE*) and noun-based AE (*NAE*). VAE extracts attributes from sentences that their verb predicates serve as an indicator for a given attribute class. NAE extracts attributes from noun-based constructions. The profile extraction method extracts six types of attributes, which include: "تاریخ تولد/ tarikhe tavallod/ date of birth","محل تولد/ mahaleh tavallod/ birth place", "مدرک/ madrak/ degree", "بستگان/ bastegan/ relatives", "شغل/ shoghl/ occupation" and "ملیت/ melliat/ nationality". These attributes are those that make the most distinctions between person entities. Fig. 2 shows the entity profiling results for a sample sentence.

محمود حسابی در سال 1281 در تهران زاده شد. **Translate:** Mahmoud Hessabi was born in Tehran in 1281.

(a)

نام شخص: محمود حسابی
محل تولد: تهران تاریخ تولد: 1281

(b)

Fig. 2. Profile extraction applied on a sample sentence; (*a*) sample sentence; (*b*) profile for person "محمود حسابی".

Profile enrichment

The aim of this stage is to augment the profiles with semantic information extracted from BabelNet ontology [16]. Profile enrichment includes two steps: (*i*) entity linking, and (*ii*) ontological attribute retrieval.

Let *e* to be the target entity that needs to be linked to an entry in the BabelNet ontology. In entity linking step, target entity *e* is disambiguated to identify which of its senses is given in the text. To do this, first the intra-document co-reference chain of the entity mention *e* is identified using a rule-based co-reference resolution method. For each of the mentions in co-reference chain of entity *e*, the sentences containing mentions are combined to create a summarized text *S*. Then, the content *S* are given to Babelfy [28], a state-of-the-art entity linking and word sense disambiguation system to obtain a sense mapping from surface text mention *e* to an ontological sense. The synset offset generated by Babelfy is mapped to possible DBPedia[3] URI. By this way, the surface entity mention is related to their corresponding meaningful identity in an online ontology. In ontological attribute retrieval step, the global attributes for the target entity *e* are retrieved from an online knowledge base. These attributes are beyond the discourse profile. The attribute extraction phase takes as input the DBPedia

[3] http://wiki.dbpedia.org/

URI of the target entity *e*, and retrieves its attribute from DBPedia ontology [29] to enrich discourse profile of entity *e*. Fig. 3 shows the semantic enrichment result for the example profile given in Fig. 2(b). In Fig. 3, notation bn:*i*n refers to the *i*th BabelNet sense for the given word. In Fig. 3, the local attributes are shown in black colour and ontological attributes obtained by profile enrichment are shown in blue colour. The BabelNet senses are shown inside brackets and DBPedia URIs are shown inside parentheses.

Babelfy may produce some noisy data because in some cases it cannot infer the correct identity of entities. Therefore, to avoid dependency on the output of the Babelfy, it is better to rank the candidate external entities and prune out candidates with low confidence. This is considered as one of the future works and in current implementations, we rely on the Babelfy itself to identify the correct identity of the entity in question.

نام شخص: محمود حسابی [bn:00838012n] (http://dbpedia.org/resource/Mahmoud_Hessaby)
محل تولد: تهران [bn:00015553n] (http://dbpedia.org/resource/Tehran) تاریخ تولد: 1281 ملیت: ایرانی (http://dbpedia.org/resource/Iran) درگذشت: 1371

Fig. 3. Profile enrichment result for the sample profile given in Fig. 2 (b).

Profile clustering

Profile clustering takes as input the augmented profiles produced by profile enrichment phase. It then groups the profiles referring to the same person in reality. Given $P=\{P_1, P_2, \ldots, P_n\}$ be the set of profiles, the goal of clustering is to group profiles into clusters $G=\{C_1, C_2, \ldots, C_k\}$, such that $k \le n$, $C_i \cap C_j = \varnothing$, for $i \ne j$, and profiles $\{P_p^i, P_{p+1}^i, \ldots, P_q^i\}$, $(1 \le p \le q \le n)$ within each cluster C_i are relevant to each one other and refer to the same person in reality. The solution to this problem by considering the notion that we do not know the number of clusters previously is to use hierarchical clustering.

In our implementations, we adopt agglomerative clustering [30] to group profiles. It is a bottom up hierarchical clustering algorithm, in which, each data object starts in its own cluster, and pairs of clusters are merged as one move up the hierarchy. Since at the starting of clustering, each profile $P_i \in P$ is in its own cluster $C_i \in C$, the number of initial clusters is equal to the number of profiles, i.e.

$|G| = |P|$. We introduced an efficient similarity measure to compute similarity among data objects. Our similarity metric relies on the closeness centrality metric, particularly closeness of personal attributes. The main idea behind our similarity metric is that "two people are closely related and can be grouped in the same cluster, if they share more common attributes." The proposed similarity measure is defined as follows:

$$O\left(C_i, C_j\right) = \frac{S\left(C_i, C_j\right)}{|C_i| \times |C_j|} \times \lambda\left(C_i, C_j\right) \tag{3}$$

$|C_i|$ and $|C_j|$ indicates the size of cluster *i* and *j*, respectively, $\lambda\left(C_i, C_j\right)$ is the sparseness penalty and $S\left(C_i, C_j\right)$ is the similarity function, which measures the normalized similarity between cluster C_i and C_j. We run the agglomerative clustering with the similarity measure defined in Eq. (3) over all clusters. $S\left(C_i, C_j\right)$ is defined as follows:

$$S(C_i, C_j) = \frac{1}{\left|A_{C_i, C_j}\right|} \times \sum_{a \in A_{C_i, C_j}} \omega_a \times M_a(C_i, C_j) \tag{4}$$

$M_a(C_i, C_j)$ is the similarity of two clusters C_i and C_j based on attribute *a*, ω_a is the importance coefficient of attribute *a*. In current implementations of this article, we set ω_a to 1 for all attributes. A_{C_i, C_j} represents all the attributes are in both clusters C_i and C_j. A_{C_i, C_j} is equal to $A_{C_i, C_j} = A_{C_i} \cup A_{C_j}$, where A_{C_i} and A_{C_j}, respectively represent the set of attributes associated with cluster C_i and C_j. In general, each attribute class $a \in A$ may be one of the following types: single-value attribute or multiple-value attribute. Single-value attribute (e.g., *date of birth*) can only take a single value, while multiple-value attribute (e.g., *affiliation* and *occupation*) can take one or more values. If an attribute *a* is a multiple-value attribute, to calculate $M_a(C_i, C_j)$, first single-value similarities are calculated for all the possible values of *a*, and then aggregate the maximum single-value similarities. $M_a(C_i, C_j)$ is defined as follows:

$$M_a(C_i, C_j) = \frac{1}{\min\left(\left|I_{C_i}\right|, \left|I_{C_j}\right|\right)} \times \sum_{p \in I_{C_i}} \max\left(\delta\left(p, I_{C_j}\right)\right) \tag{5}$$

I_{C_i} and I_{C_j} represent the item set of attribute a for cluster C_i and C_j, respectively. $\delta\left(p, I_{C_j}\right)$ is the set of single-value attribute similarities computed between element $p \in I_{C_i}$ and all elements in I_{C_j}. We define $\delta\left(p, I_{C_j}\right)$ as follows:

$$\delta\left(p, I_{C_j}\right) = \left\{\varphi(p, q) \middle| q \in I_{C_j}\right\} \tag{6}$$

$\varphi(p,q)$ calculates the similarity between item *p* and q using an appropriate standard similarity measure. Personal attributes are heterogeneous; therefore, it isn't reasonable to use the same similarity measure for different attributes in computing $\varphi(p,q)$. This enforces us to use appropriate similarity measures for any type of the attribute. There are different standard similarity measures, each of which is appropriate for a particular attribute class. To compute similarity between attribute profiles, we used four types of similarity measures: Levenshtein distance [31], Cosine [32], Dice's coefficient [32] and dates' relative similarity (*Spd*) [33].The reason to select these measures is that these are widely used in literature to calculate similarity of data objects. In our implementations, we empirically used normalized Levenshtein distance for the attributes of degree, nationality and occupation; the Cosine similarity metric for the attribute of relatives; the Dice coefficient for the attribute of birth place; the *Spd* measure for the attribute date of birth.

We apply the similarity measures on single-value items of attributes. Each similarity measure has its own strategy to compute similarity value. For example, to compute similarity by Cosine measure, we first transform the single-value items to vectors of occurrences of n-grams (sequences of *n* characters). In this *n*-dimensional space, the similarity between two items is the cosine of their respective vectors. In other words, it is computed as $(V_1.V_2)/(|V_1|\times|V_2|)$, where V_1 and V_2 is the vector representation of two comparing items *p* and *q*. Borrowing the idea presented in [33], to compare *date of birth* values, we first convert dates into a number of days. We calculate the number of days according to the fix date 01-01-1395. Let d_1 and d_2 be the two day values that are being compared, *Spd*, the dates' relative similarity is calculated as follows:

$$Spd\,(d_1,d_2)=\begin{cases}1-\left(\dfrac{pd\,(d_1,d_2)}{pd_{\max}}\right) & \text{if } (pd\,(d_1,d_2)<pd_{\max}) \\ 0 & \text{else}\end{cases} \tag{7}$$

$pd_{\max}$ ($0<pd_{\max}<1$) is the maximum percentage difference that is tolerated in similarity computation. In our implementations, we empirically set $pd_{\max}$ to 0.2. $pd\,(d_1,d_2)$ is the percentage difference, which is defined as follows:

$$pd\,(d_1,d_2)=\frac{|d_1-d_2|}{\max(d_1,d_2)} \tag{8}$$

Obviously, the bigger the $S(C_i,C_j)$ value is, the higher the cluster closeness is. In clustering algorithm, merging decisions use the single link score between all new links across two clusters. $\lambda(C_i,C_j)$ is the sparseness penalty and it is defined as follows :

$$\lambda\left(C_i, C_j\right) = \frac{\sum_{n_i \in C_i} \sum_{n_j \in C_j} 1\left\{R\left(n_i,\ n_j\right) > 0\right\}}{\sum_{n_i \in C_i} \sum_{n_j \in C_j} 1} \tag{9}$$

This penalty controls the attribute density between clusters C_j and C_j, and penalizes clusters from merging when they share only a few attributes.

We continue merging clusters until the minimum combination similarity between clusters is greater than an empirical stopping threshold. The proper setting of the stopping threshold for agglomerative clustering algorithm has a great impact to the quality and robustness of a personal name disambiguation method. We identify the stopping threshold using test and trail method. In this method, we ran the clustering with all possible stopping thresholds (starting from 0 to 1 with step 0.05) on benchmark dataset and the one with the best B-cubed F-score is selected for each personal name. We then ran the system with the fixed optimal stopping threshold on the dataset.

4. Experiments and results

In this section, we first describe benchmark datasets and performance metrics and then give the results obtained by our approach and baseline methods.

Datasets

1) Farsi dataset

A key challenge to evaluate our proposed approach is the lack of Farsi dataset suited for name disambiguation problem. To solve this issue, we created a small Farsi name disambiguation corpus for evaluating our approach. We first choose 10 different Farsi person names and then queried the web for these names. The name list included several common names like 'احمد عبدالهزاده/ Ahmad Abdollahzadeh'. The reason to using personal names as benchmark comes from the fact that searching for web pages related to a person is a common activity in current web searches. For each name, at most top 20 web pages returned by Google[4] search engine are collected for disambiguation and included into the dataset. In total, there are 200 web pages in dataset. This dataset provides a real corpus, which can test a disambiguation system for personal names with varying ambiguity and in different domains. To create ground truth, we asked two human annotators to independently label web pages for each query name. The annotators reached an agreement score of $\kappa = 76\%$ measured by Cohen's kappa coefficient [34], which considered to be within the substantial agreement boundaries. During

[4] www.google.com

the annotation process, the annotators identified 54 different individuals, each of which refers to a cluster.

2) English dataset

In order to make a fair evaluation, we used two English dataset to validate and compare our work with other approaches. These datasets are WePS-1 test dataset[5] [35], and WePS-2 test dataset[6] [2]. Each dataset contains collections of web pages obtained from the results for a personal name query to a search engine. Both WePS-1 test and WePS-2 test datasets consists of 30 personal names and for each name, the top ranked *N* web pages (100 for WePS-1 and 150 for WePS-2) from the search results were included into the datasets. These datasets provide a real corpus, which can test a disambiguation system for personal names with varying ambiguity and in different domains. The personal names were chosen from three different sources (10 name sets from Wikipedia, 10 from the US Census, and 10 from ACL conference) in order to provide different ambiguity scenarios. For each dataset the ground truth files are also provided by expert annotators.

Performance measures

Various measures are presented to evaluate the quality of name disambiguation algorithms. We conducted evaluations using B-cubed scoring measure [35, 36]. We use three B-cubed scoring measures including precision, recall and F-score. A more detailed discussion of these quality metrics is given in [35, 36].

The system's performance is measured by comparing the clustering generated by system with human labeled gold-standard test data. Let $E = \{e_1, e_2, \ldots, e_n\}$ to be the set of entities in the dataset, $R = \{R_1, R_2, \ldots, R_k\}$ to be the clusters generated by the system and $G = \{G_1, G_2, \ldots, G_m\}$ to be the clusters of the annotated gold-standard. B-cubed precision (B^3P) is defined as follows:

$$B^3P = \frac{\sum_i P(e_i)}{|E|} \tag{10}$$

$$P(e_i) = \frac{|R_{k_i} \cap G_{k_i}|}{|R_{k_i}|} \tag{11}$$

Where $|E|$ is the number of clustered items, and $P(e_i)$ is the precision score of entity mention e_i, $R_{k_i} \in R$ is the *k*th cluster in system output, which includes

---

[5] Available for download at [http://nlp.uned.es/weps/weps-1/weps1-data]
[6] Available for download at [http://nlp.uned.es/weps/weps-2/weps2-data]

the entity mention e_i, and $G_{k_i} \in G$ is the *k*th cluster in annotated gold standard with e_i.

B-cubed recall (B^3R) is defined as follows:

$$B^3R = \frac{\sum_i R(e_i)}{|N|} \tag{12}$$

$$R(e_i) = \frac{|R_{k_i} \cap G_{k_i}|}{|G_{k_i}|} \tag{13}$$

where $R(e_i)$ is the recall of entity mention e_i.

B-cubed F-score $(B^3\mathrm{F}_\alpha)$, which is the harmonic mean of B-cubed precision and recall is defined as follows:

$$B^3\mathrm{F}_\alpha = \frac{1}{\alpha\frac{1}{B^3P} + (1-\alpha)\frac{1}{B^3R}} \tag{14}$$

In addition to B-cubed measures, we used three purity-based scoring measures, including purity (*Pr*), inverse purity (*IPr*), and harmonic mean of purity and inverse purity (F_p-score).

5. Numerical results and discussion

We perform all experiments on a 3GHz, and 4GB RAM Personal Computer Intel Pentium 4. We coded all the mentioned algorithms using Java and MATLAB language.

1) Experiments on Farsi dataset

Table 1 shows the results obtained by our name disambiguation method on Farsi dataset. As shown in Table 1, our proposed method achieved 57.2% B-cubed precision, 68.6% B-cubed recall and 61.93% B-cubed F-score. However, the performances are far from being ideal. This shows that name disambiguation in Farsi textual resources is a big challenge and justifies that more effort is needed in this field.

Table 1

Performance of our method on Farsi dataset

Method	B^3P (%)	B^3R (%)	$B^3\mathrm{F}_{\alpha=0.5}$ (%)
Our method	57.2	68.6	61.93

In order to show the impact of each clustering feature, in Table 2, we begin with the feature of “local discourse profile attributes” and then append “ontological profile attributes”. The results clearly show the impact of profile information enrichment and integrating local and ontological attributes. In Table 2, we notice that the performance is increasing when incorporating ontological attributes. The final feature model (local discourse attributes + ontological attributes) achieves the best performances.

Table 2

Performance of our method using different features on Farsi dataset

Method	B^3P (%)	B^3R (%)	$B^3F_{\alpha=0.5}$ (%)
Local profile attributes	51.21	61.4	55.38
+ Ontological profile attributes	57.2	68.6	61.93

We implemented four name disambiguation methods as our baseline methods. These baseline methods include: (*i*) bag of words model (BOW model) [14], (*ii*) attribute-based method (AV model) [12, 18], (*iii*) ALL-IN-ONE [2] and (*iv*) ONE-IN-ONE method [2]. The BOW baseline [14] is based on the traditional bag of words models: agglomerative vector space clustering with TF/IDF weighting schema. The BOW method is widely employed as a benchmark in a series of previous work. The AV baseline [12, 18] is an attribute-based name disambiguation algorithm, which relies only on the personal attributes. The ALL-IN-ONE and ONE-IN-ONE baselines are provided by the WePS sharetask [2]. In ALL-IN-ONE baseline all documents related to a person are placed in a single cluster. In contrast, in ONE-IN-ONE baseline each document is included in a separate cluster. We notice that, we implemented the baseline methods as described in their original paper. Table 3 summarizes the results obtained by baseline methods and our name disambiguation method on the Farsi benchmark dataset.

Comparing to baseline methods, our method clearly outperformed the baseline methods. Our method achieves higher overall $B^3F_{\alpha=0.5}$ score, 2.76% better than BOW, 6.55% better than AV [18], 3.5% better than ALL-IN-ONE [18] and 20.91% better than ONE-IN-ONE. This indicates that our approach increases the performance of name disambiguation. The ONE-IN-ONE baseline obtained the best result in terms of B^3P measure. The ALL-IN-ONE baseline outperformed other algorithms in terms of B^3R measure. The higher B^3R for ALL-IN-ONE baseline arises from the fact that in benchmark datasets, half of the documents belong to one specific person. Performance order in terms of B-cubed precision is ONE-IN-ONE> our method> BOW> AV> ALL-IN-ONE. In terms of B-cubed recall the performance order is ALL-IN-ONE> BOW> our method> AV> ONE-IN-ONE. The performance order in terms of B-cubed F-score is our method> BOW> ALL-IN-ONE> AV> ONE-IN-ONE.

Table 3

Comparison of results obtained by baselines and our method on Farsi dataset

Method	B^3P (%)	B^3R (%)	$B^3F_{\alpha=0.5}$ (%)
BOW	53.9	70.2	59.17
AV	51.21	61.4	55.38
ALL-IN-ONE	47.3	78.9	58. 43
ONE-IN-ONE	93.05	26.3	41.02
Our method	57.2	68.6	61.93

Table 4 represents the comparison of our method and the state-of-the-art method presented in [18]. The results show that, our approach performs well and exceeding the performance obtained by the state-of-the-art method in terms of performance measures.

Table 4

Comparison of results obtained by state-of-the-art and our method on Farsi dataset

Method	B^3P (%)	B^3R (%)	$B^3F_{\alpha=0.5}$ (%)
Emami et al. [18]	52.43	62.97	56.17
Our method	57.2	68.6	61.93

2) Experiments on English dataset

Table 5 shows the results obtained by our name disambiguation method on WePS-1 test dataset. Table 6 shows the results for WePS-2 test dataset. In order to indicate the effect of each clustering feature, in Tables 5 and 6, we begin with the feature of "*local discourse profile attributes*" and then add features of local "*ontological profile attributes*". The results clearly show the effect of profile enrichment and integrating local profile attributes with ontological attributes. The final feature model *local attributes* + *ontological attributes* achieves the best performances. In Table 1, the performances on WePS-1 test dataset increase about +10.07% in terms of $B^3F_{\alpha=0.5}$ and from the local feature model *local attributes* to final feature model *local attributes* + *ontological attributes*. The improvement rate from the local feature model to final feature model is about +7.45% in terms of $B^3F_{\alpha=0.5}$ for WePS-2 test dataset. This justifies that the majority of information for name disambiguation is given in the local web pages being processed. However, incorporating the ontological attributes improves the performance of name disambiguation.

Table 5

Performances of our name disambiguation method on WePS-1 test dataset

Method	B^3P (%)	B^3R (%)	$B^3F_{\alpha=0.5}$ (%)
Local profile attributes	58.42	67.22	61.97
+ ontological profile attributes	68.35	76.37	72.04

Table 6

Performances of our name disambiguation method on WePS-2 test dataset

Method	B^3P (%)	B^3R (%)	$B^3F_{\alpha=0.5}$ (%)
Local profile attributes	63.23	76.30	66.15
+ ontological profile attributes	68.49	81.91	73.60

We compared our approach with five baseline methods that include: (*i*) bag of words model (BOW model) [14], (*ii*) social network based method (SN model) [23], (*iii*) attribute-based method (AV model) [12], [19], (*iv*) ALL-IN-ONE [2] and (v) ONE-IN-ONE method [2]. Table 7 shows the best performance obtained by the baselines and our method on WePS-1 test dataset. Table 8 shows the results for WePS-2 test dataset. As shown in Table 7 and 8, our method clearly outperforms the baseline methods for both datasets in terms of $B^3F_{\alpha=0.5}$. For WePS-1, on average our method outperforms BOW, SN, AV, ALL-IN-ONE, and ONE-IN-ONE by +2.74%, +3.44%, +11.34%, +16.04% and +40.04%, respectively in terms of $B^3F_{\alpha=0.5}$. The improvement is also evident for WePS-2 dataset, in which our method obtains +1.1%, +6.3%, +12.2%, +20.6% and +39.6% improvement compared to BOW, SN, AV, ALL-IN-ONE, and ONE-IN-ONE, respectively, in terms of $B^3F_{\alpha=0.5}$. The ONE-IN-ONE baseline obtained the best result in terms of B^3P measure on both WePS-1 and WePS-2 datasets. The ALL-IN-ONE baseline outperformed other algorithms in terms of B^3R measure. The higher B^3P for ONE-IN-ONE baseline arises from the fact that in WePS-1 andWePS-2 datasets documents are distributed among the clusters. Since in average half of the documents in the dataset belong to one specific person, the ALL-IN-ONE baseline gave better results in terms of B^3R .

Table 7

Comparison of results obtained by baselines and our method on WePS-1 test dataset

Method	B^3P (%)	B^3R (%)	$B^3F_{\alpha=0.5}$ (%)
BOW	62.1	75.5	69.3
SN	65.0	73.5	68.6
AV	59.4	68.4	60.7
ALL-IN-ONE	44.0	100	56.0
ONE-IN-ONE	100	20.0	32.0
Our method	68.35	76.37	72.04

Table 8

Comparison of results obtained by baselines and our method on WePS-2 test dataset

Method	B^3P (%)	B^3R (%)	$B^3F_{\alpha=0.5}$ (%)
BOW	66.2	80.5	72.5
SN	64.2	81.1	67.3
AV	62.5	77.7	61.4
ALL-IN-ONE	43.0	100	53.0
ONE-IN-ONE	100	24.0	34.0
Our method	68.49	81.91	73.60

Our method still suffers from several challenges that need to be addressed. Our manual investigation over incorrect classified objects indicates that the performance for name disambiguation can be raised if the following conditions are hold.

- ***Improving the performance of pre-requisite components*:** our manual investigation reveals that almost half of the incorrect classified web pages were because of the inefficiency of pre-processing and profile extraction components, and not because of the inefficiency of our name clustering method. Errors in pre-requisite stages are propagated to name clustering step and cause wrong grouping. Thus, the low performance of pre-requisite stages is a bottleneck for efficient name disambiguation. However, improving pre- requisite is orthogonal to our problem and therefore out of the scope of this article. To alleviate the errors in pre-processing and profile extraction stage, we utilize the profile enrichment strategy and enrich discourse profiles with semantic information extracted from an online ontology. Table 2 clearly shows the positive impact of knowledge enrichment on name disambiguation.
- ***Incorporating more features for name disambiguation:*** in this article, the clustering features rely only on the discourse profile attributes and ontological features. Our approach ignores some other semantic discourse features such as social relations among person entities. Exploiting all of the information contained in given web pages including social links can improve the efficiency of name disambiguation.

6. Conclusion

In this paper, we have investigated the problem of personal name disambiguation across Farsi web pages. We have proposed a combination approach to deal with the problem. We formalized the name disambiguation problem as four main subtasks: pre-processing, profile extraction, profile enrichment and profile clustering. Specifically, we have proposed a semantic knowledge enrichment approach to augment the discourse profile information of individuals in question through linking named entities to an online multi-lingual ontology and then extracting their relevant semantic knowledge. Further, we have defined an efficient similarity measure and run the agglomerative clustering algorithm with the proposed similarity measure to cluster enriched profiles and disambiguate names across web pages. We evaluated the proposed method on a Farsi name disambiguation corpus. Experimental results indicate that our proposed method clearly outperforms several baseline methods.

On the whole, our name disambiguation approach can be considered as a foundation for more robust name disambiguation approaches in Farsi. Our proposed approach is specialized to work with web content written in Farsi, but it can be easily generalized to work with web content in other languages, too. There are several potential enhancements of this work. First, we plan to design a generic name disambiguation system to cover more types of entities and accurately disambiguate various types of entities. As the final results of name disambiguation system depend on the performance of two pre-requisite subtasks including pre-processing and profile extraction, therefore second interesting future work is to improve the pre-requisites' performance, which eventually can improve the overall quality of name disambiguation system. Since the problem of name disambiguation is far from being solved, our third future work is to exploit all of the information contained in local web pages especially social relations among entities. Finally, we would like to work on algorithms for multilingual cross-document name disambiguation, which aims to disambiguate personal names across multilingual web pages.

REFERENCES

[1] A. A. Barforoush, H. Shirazi, and H. Emami, "A new classification framework to evaluate the entity profiling on the Web: past, present and future," ACM Comput. Surv., vol. 50, no. 3, pp. 1–39, 2017.

[2] 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), 18th WWW Conference, Madrid, Spain, 2009.

[3] H. Emami, H. Shirazi, and A. A. Barforoush, "A Semantic approach to person profile extraction from Farsi documents," J. Inf. Syst. Telecommun., vol. 4, no. 4, pp. 232–243, 2016.

[4] N. E. Jivan, M. Fazeli, and K. S. Yousefi, "New approach for automated categorizing and finding similarities in online Persian news," in International Symposium on Information Management in a Changing World. Springer, Berlin, Heidelberg, 2010, pp. 120–128.

[5] M. Farhoodi, A. Yari, and M. Mahmoudi, "Combining Content-Based and Context-Based Methods for Persian Web Page Classification," in Applications of Digital Information and Web Technologies, 2009. ICADIWT'09. Second International Conference on the, 2009, pp. 399–404.

[6] M. Farhoodi, "Quality Improvement of Information Retrieval System Results by Document Classification," Int. J. Inf. Commun. Technol. Res., vol. 5, no. 3, 2013.

[7] M. Fathian and A. M. Karimi-Majd, "A new algorithm for clustering web-pages based on links and content," Sharif J. Ind. Eng. Manag., vol. 33, no. 1, pp. 21–28, 2017.

[8] A. Bagga and B. Baldwin, "Entity-based cross-document coreferencing using the Vector Space Model," in Proceedings of COLING-ACL-17, Montral, Qubec, Canada, 1998, pp. 79–85.

[9] S. R. Yerva, Z. Miklós, and K. Aberer, "Quality-aware similarity assessment for entity matching in Web data," Inf. Syst., vol. 37, no. 4, pp. 336–351, 2012.

[10] Y. Chen, S. Y. Mei Lee, and C. R. Huang, "A robust web personal name information extraction system," Expert Syst. Appl., vol. 39, no. 3, pp. 2690–2699, 2012.

[11] G. S. Mann and D. Yarowsky, "Unsupervised personal name disambiguation," in Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003, Edmonton,

Canada, 2003, pp. 33–40.
[12] H. Srinivasan, J. Chen, and R. Srihari, "Cross document person name disambiguation using entity profiles," in Proceedings of the Second Text Analysis Conference (TAC 2010), Maryland, USA, 2009.
[13] D. V Kalashnikov, Z. Chen, S. Mehrotra, and R. Nuray-Turan, "Web People Search via Connection Analysis," IEEE Trans. Knowl. Data Eng., vol. 20, no. 11, pp. 1550–1565, 2008.
[14] X. Han and J. Zhao, "Structural Semantic Relatedness: A Knowledge-Based Method to Named Entity Disambiguation," in Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 2010, pp. 50–59.
[15] S. Dutta and G. Weikum, "Cross-document co-reference resolution using sample-based clustering with knowledge enrichment," Trans. Assoc. Comput. Linguist., vol. 3, pp. 15–28, 2015.
[16] R. Navigli and S. P. Ponzetto, "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," Artif. Intell., vol. 193, pp. 217–250, 2012.
[17] U. Brandes and T. Erlebach, Network analysis: methodological foundations. Springer Science & Business Media, 2005.
[18] H. Emami, H. Shirazi, A. A. Barforoush, and M. Hourali, "A Pattern-Matching Method for Extracting Personal Information in Farsi Content," U.P.B. Sci. Bull., Ser. C, vol. 78, no. 1, pp. 125–138, 2016.
[19] I. Nagy, "Person attribute extraction from the textual parts of Web pages," Acta Cybern., vol. 20, no. 3, pp. 419–440, 2012.
[20] M. Lan, Y. Z. Zhang, Y. Lu, J. Su, and C. L. Tan., "Which Who are They? People Attribute Extraction and Disambiguation in Web Search Results," In2nd Web People Search Eval. Work. (WePS 2009), 18th WWW Conf. Madrid, Spain, 2009.
[21] F. Song, R. Cohen, and S. Lin., "Web People Search Based on Locality and Relative Similarity Measures," in 2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference, Madrid, Spain, 2009.
[22] R. Bekkerman and A. Mccallum, "Disambiguating Web appearances of people in a social network," in WWW '05: Proceedings of the 14th international conference on World Wide Web, Chiba, Japan, 2005, pp. 463–470.
[23] B. Malin, "Unsupervised Name Disambiguation via Social Network Similarity," in Proceedings of the Workshop on Link Analysis, Counterterrorism, and Security, in conjunction with the SIAM International Conference on Data Mining. Newport Beach, CA, 2005, vol. 1401, pp. 93–102.
[24] X. Han and J. Zhao, "Named entity disambiguation by leveraging wikipedia semantic knowledge," in Proceeding of the 18th ACM conference on Information and knowledge management-CIKM '09, Hong Kong, China, 2009, pp. 215–224.
[25] Y. Li, C. Wang, F. Han, J. Han, D. Roth, and X. Yan, "Mining evidences for named entity disambiguation," in Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, Illinois, USA, 2013, pp. 1070–1078.
[26] H. Emami, H. Shirazi, and A. A. Barforoush, "A semantic approach to cross-document person profiling in Web," J. Inf. Syst. Telecommun., vol. 4, no. 4, pp. 232–243, 2016.
[27] R. Al-Rfou, V. Kulkarni, B. Perozzi, and S. Skiena, "Polyglot-NER: Massive Multilingual Named Entity Recognition," in Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, British Columbia, Canada, 2015, pp. 586–594.
[28] A. Moro, A. Raganato, and R. Navigli, "Entity linking meets word sense disambiguation : a unified approach," Trans. Assoc. Comput. Linguist., vol. 2, pp. 231–244, 2014.
[29] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A nucleus

for a Web of open data,” Proc. 6th Int. Semant. Web Conf. (ISWC), Busan, Korea, vol. 4825 LNCS, pp. 722–735, 2007.
[30] L. Rokach and O. Maimon, “Clustering methods,” in Data mining and knowledge discovery handbook, Springer US, 2005, pp. 321–352.
[31] L. Yujian and L. Bo, “A normalized Levenshtein distance metric,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 6, pp. 1091–1095, 2007.
[32] C. Liu, “Discriminant analysis and similarity measure,” Pattern Recognit., vol. 47, no. 1, pp. 359–367, 2014.
[33] P. Christen, Data Matching Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection. Springer Science & Business Media, 2012.
[34] J. R. Landis and G. G. Koch, “The measurement of observer agreement for categorical data,” Biometrics, vol. 33, no. 1, pp. 159–174, 1977.
[35] J. Artiles, J. Gonzalo, and S. Sekine, “The SemEval-2007 WePS Evaluation: Establishing a benchmark for the Web People Search Task,” in Proceedings of the 4th International Workshop on Semantic Evaluations, (SemEval-2007), Prague, Czech Republic, 2007, pp. 64–69.
[36] E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo, “Evaluation metrics for clustering tasks: a comparison based on formal constraints,” Inf. Retr. Boston., vol. 12, no. 4, pp. 461–486, 2007.
[37] J. Cai and M. Strube, “Evaluation metrics for end-to-end coreference resolution systems,” in Proceedings of SIGDIAL 2010: the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue, Tokyo, Japan, 2010, pp. 28–36.