Improving Wikipedia-based Place Name Disambiguation in Short Texts Using Structured Data from DBpedia

Yingjie Hu
STKO Lab
University of California, Santa Barbara, USA
yingjiehu@geog.ucsb.edu

Krzysztof Janowicz
STKO Lab
University of California, Santa Barbara, USA
jano@geog.ucsb.edu

Sathya Prasad
Application Prototype Lab
Esri Inc., Redlands, USA
sprasad@esri.com

ABSTRACT

Place name disambiguation is an important task for improving the accuracy of geographic information retrieval. This task becomes more challenging when the input texts are short. Wikipedia provides information about places and has often been employed for named entity recognition. However, the natural language representation of Wikipedia articles limits more effective use of this rich knowledge base. DBpedia is the Semantic Web version of Wikipedia, which provides structured and machine-understandable knowledge mined from Wikipedia articles. This paper presents an approach for combining Wikipedia and DBpedia to disambiguate place names in short texts. We discuss the pros and cons of the two knowledge bases, and argue that a combination of both performs better than each of them alone. We evaluate our proposed method by conducting experiments against baselines of three established methods. The result indicates that our method has a generally higher precision and recall. While our study employs DBpedia, the proposed method is generic and can be extended to other structured Linked Datasets such as Freebase or Wikidata.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: [Information Search and Retrival]; I.2.7 [**Artificial Intelligence**]: [Natural Language Processing]

Keywords

Place name disambiguation, Wikipedia, DBpedia, Linked Data

1. INTRODUCTION

Place name disambiguation is an important task for improving the accuracy of geographic information retrieval (GIR) [10]. This importance can be reflected in two components of GIR. The first one is the query pre-processing. Given a text query from the user, the place search engine should be able to figure out the geographic entity that the user is referring to in order to return relevant information. The second component is the indexing of data records in the target database. Better search results can be achieved, if the search engine can disambiguate the geographic entities mentioned in the natural language descriptions of data records, and establish an additional index based on the recognized entities [5].

While important for GIR, place name disambiguation also presents challenges. A key problem is the ambiguity of toponyms [4, 17]. The same place name can refer to different geographic entities (e.g., the Getty Thesaurus of Geographic Names returns 102 places for the toponym *Santa Barbara*; many of them being populated places). Additionally, the same geographic entity can have different names (e.g., the city of Istanbul in Turkey has at least 12 known toponyms). A common strategy to handle this challenge is to employ the surrounding words as context information. Such context information is then compared to the ground truth descriptions of the places to be disambiguated, and the place with highest similarity is returned as the result.

Wikipedia is a knowledge base that has often been used as the source for ground truth descriptions [17, 3, 6, 14]. Due to its plain text nature, Wikipedia articles have often been reduced to vector space models, in which the importance of a term is based on its frequency and inverse document frequency (TF-IDF). For example, Bunescu and Pasca used TF-IDF and cosine similarity to compare the context information of an entity with Wikipedia articles [3]. Using the same metric, Cucerzan enriched the term vectors by including not only words but also short phrases [6]. Other similar approaches have also been proposed in the literature that aim at enriching the term vectors derived from Wikipedia articles [15, 16].

While a vector space model can often achieve fair disambiguation results, it ignores the semantic importance of terms. To give a concrete example, consider the following sentence: *"Washington was first established in 1824, and it is home to the Historic Washington State Park"*. In this sentence, the term *Historic Washington State Park* provides important clues about the place. An individual familiar with this park may correctly infer that this sentence is about *Washington, Arkansas.* However, a vector space model will treat this phrase as equal to other words (e.g., *established* and *home*) or may even divide it into four terms. Note that the toponym "Historic Washington State Park" only appears once on the Wikipedia page of Washington, Arkansas[1].

GIR'14, November 04 2014, Dallas, TX, USA
Copyright 2014 ACM 978-1-4503-3135-7/14/11 ...$15.00.

[1] http://en.wikipedia.org/wiki/Washington,_Arkansas. Ac-

Thus, *semantically indicative* terms do not necessarily have high frequencies in Wikipedia articles. This significantly reduces their *term frequency* value and consequently decreases their importance in a vector space representation.

Identifying semantically indicative terms is even more important for place disambiguation in short texts. Short text, such as user queries or snippets published together with data records, only contain very limited context information. In such situations, recognizing just one important clue can already improve the disambiguation accuracy. These clues can include higher-level administrative units (e.g., states and counties) that this place belongs to, nearby places (e.g., neighboring communities), famous people associated with the place, characteristic landmarks, and so forth. An intuitive way to incorporate these clues into place disambiguation is to make use of the hyperlinks within Wikipedia articles. While some hyperlinks do point to entities that are directly related to the entity described by the article, there are also many geographically unrelated links, since Wikipedia authors are encouraged to create a dense network of interlinked articles. Figure 1 shows a fragment of the Wikipedia page of Washington, Arkansas. As can be seen from the figure, terms, such as *"United States Census Burean"*,*"census"*, and *"poverty line"*, are all hyperlinked, although these terms are not specific to *Washington, Arkansas* but point to general concepts.

DBpedia is a central Semantic Web hub that extracts and enriches information of Wikipedia. Based on the Resource Description Framework (RDF), DBpedia organizes Wikipedia knowledge into a structured and machine-understandable graph [11]. Focusing on *things* (entities) instead of *strings* (natural langague descriptions), DBpedia represents the relations among entities in the world. A quick check of the DBpedia page for *Washington, Arkansas*[2] reveals known facts about the city, e.g., the county and state it belongs to, its population, as well as related places such as the *Historic Washington State Park*; see Figure 2. DBpedia only lists the entities that have direct connections to the target. Thus, these entities can be used as important clues to improve place name disambiguation. Similar to Wikipedia, DBpedia also provides data in different language versions, and such a multilingual feature offers the potential to process place names in languages other than English.

In this paper, we present an approach for improving Wikipedia-based place name disambiguation in short texts using structured data from DBpedia. **The contributions of this work are as follows:**

- We analyze the pros and cons of Wikipedia and DBpedia in place name disambiguation.
- We propose a method for combining structured and unstructured data from these two data sources. We demonstrate that such a combination results in a higher disambiguation accuracy than three baseline approaches.
- We shared the source code and ground truth datasets on Github[3] to make our work reproducible and to provide a new baseline for future studies.

The remainder of this paper is organized as follows. Section 2 describes a general procedure used for place name disambiguation and discusses the pros and cons of Wikipedia and DBpedia in supporting the different stages of this procedure. Section 3 presents our method for combining the two knowledge bases. Section 4 conducts experiments to evaluate the proposed method by applying it, along with three other baseline methods, to the same datasets. Finally, section 5 summarizes this work and discusses future directions.

2. PLACE NAME DISAMBIGUATION

Similar to the process of general named entity recognition, place name disambiguation can be divided into two stages: spotting and disambiguation. This section provides some background knowledge on both stages, and discusses the roles Wikipedia and DBpedia can play. Due to the scope of this paper, our discussion will focus on methods using data from Wikipedia and DBpedia. Other data sources, such as WordNet, Getty Theauraus of Geographic Names (TGN), GEOnet Names Server (GNS), and Geographic Names Information System (GNIS) have also been used in existing works for place name disambiguation [19, 18, 5].

2.1 Stage 1: Spotting

The task of spotting is to extract the terms which can be used to represent entities in the world [17, 13]. These terms are called *surface forms*, and this stage only identifies these surface forms, and does not try to discover which entity a surface form actually refers to. For example, consider the sentence *"Greenville, the county seat of Meriwether County, became a city on December 20, 1828."*. The stage of spotting only needs to detect that *Greenville* is a surface form, and does not have to understand whether it refers to *Greenville, Georgia*, or *Greenville, Alabama*, or another *Greenville.*

To find out which terms can be used as surface forms, existing works have suggested three data sources from Wikipedia [3, 13]: Wikipedia article titles, redirect pages, and disambiguation pages. In Wikipedia, each article mainly describes one entity, and the title of an article often refers to the formal name of the described entity. Redirect pages contain information about the alternative names of one entity. For example, the wikipedia page of *United States* has corresponding redirect pages of *U.S.*, *U.S.A.*, *US*, and *USA*. Disambiguation pages provide mapping from one surface form to multiple entities. For example, the disambiguation page of the term *Washington*[4] points to 3 most prominent entities (i.e., *George Washington*, *Washington D.C.*, and *Washington State*), as well as many smaller cities and communities which are also called *Washington.* After the surface forms have been extracted, indexing tools, such as Lucene, can be employed to establish mapping links between entities and their corresponding word representations.

It is worth to note that the spotting stage does not need to use data from DBpedia. This is because DBpedia describes the same set of entities as Wikipedia does, and therefore it is unnecessary to apply the same procedure twice.

2.2 Stage 2: Disambiguation

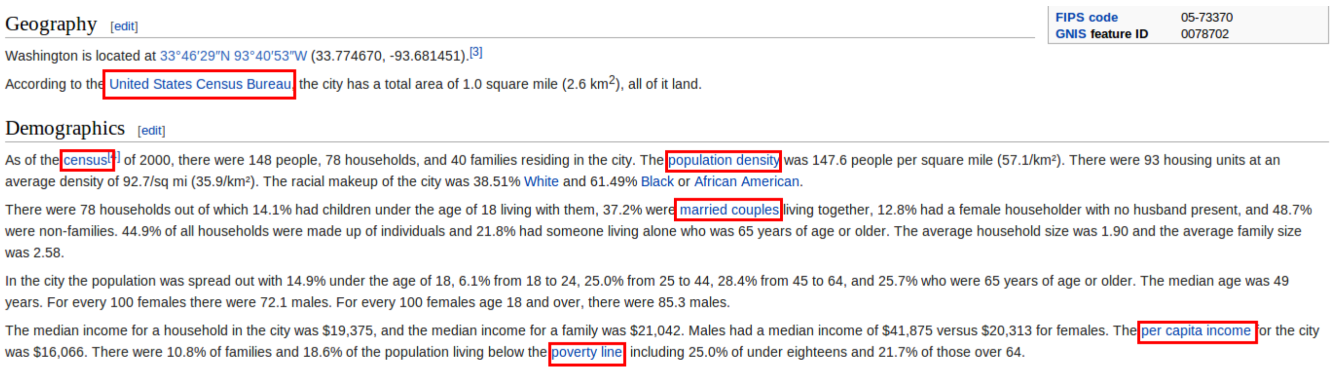
While surface forms have been extracted in stage 1, the goal of stage 2 is to identify the entity that one surface form refers to given its surrounding text context. To achieve this

cessed on August 29, 2014

[2] http://dbpedia.org/page/Washington,_Arkansas. Accessed on August 29, 2014

[3] https://github.com/YingjieHu/Place-Disambiguation

[4] http://en.wikipedia.org/wiki/Washington_(disambiguation)



FIPS code	05-73370
GNIS feature ID	0078702

Geography [edit]

Washington is located at 33°46′29″N 93°40′53″W (33.774670, -93.681451).[3]

According to the United States Census Bureau, the city has a total area of 1.0 square mile (2.6 km²), all of it land.

Demographics [edit]

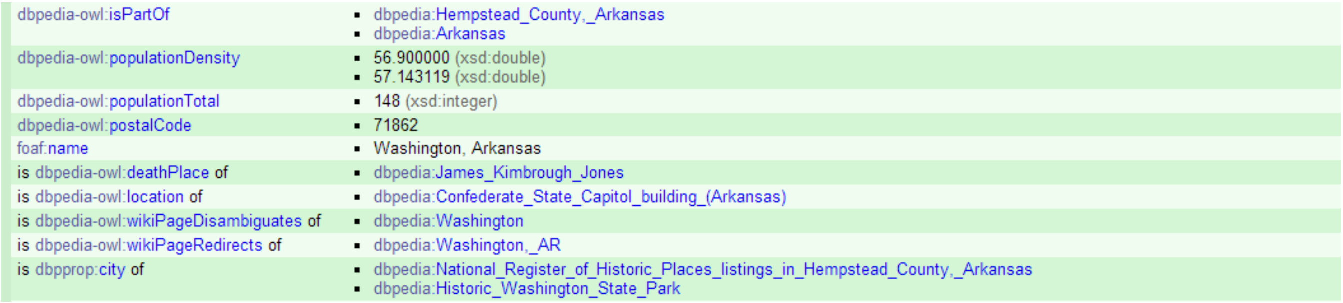
As of the census[4] of 2000, there were 148 people, 78 households, and 40 families residing in the city. The population density was 147.6 people per square mile (57.1/km²). There were 93 housing units at an average density of 92.7/sq mi (35.9/km²). The racial makeup of the city was 38.51% White and 61.49% Black or African American.

There were 78 households out of which 14.1% had children under the age of 18 living with them, 37.2% were married couples living together, 12.8% had a female householder with no husband present, and 48.7% were non-families. 44.9% of all households were made up of individuals and 21.8% had someone living alone who was 65 years of age or older. The average household size was 1.90 and the average family size was 2.58.

In the city the population was spread out with 14.9% under the age of 18, 6.1% from 18 to 24, 25.0% from 25 to 44, 28.4% from 45 to 64, and 25.7% who were 65 years of age or older. The median age was 49 years. For every 100 females there were 72.1 males. For every 100 females age 18 and over, there were 85.3 males.

The median income for a household in the city was $19,375, and the median income for a family was $21,042. Males had a median income of $41,875 versus $20,313 for females. The per capita income for the city was $16,066. There were 10.8% of families and 18.6% of the population living below the poverty line, including 25.0% of under eighteens and 21.7% of those over 64.

Figure 1: A fragment of the Wikipedia page of Washington, Arkansas.



dbpedia-owl:isPartOf	▪ dbpedia:Hempstead_County,_Arkansas ▪ dbpedia:Arkansas
dbpedia-owl:populationDensity	▪ 56.900000 (xsd:double) ▪ 57.143119 (xsd:double)
dbpedia-owl:populationTotal	▪ 148 (xsd:integer)
dbpedia-owl:postalCode	▪ 71862
foaf:name	▪ Washington, Arkansas
is dbpedia-owl:deathPlace of	▪ dbpedia:James_Kimbrough_Jones
is dbpedia-owl:location of	▪ dbpedia:Confederate_State_Capitol_building_(Arkansas)
is dbpedia-owl:wikiPageDisambiguates of	▪ dbpedia:Washington
is dbpedia-owl:wikiPageRedirects of	▪ dbpedia:Washington,_AR
is dbpprop:city of	▪ dbpedia:National_Register_of_Historic_Places_listings_in_Hempstead_County,_Arkansas ▪ dbpedia:Historic_Washington_State_Park

Figure 2: A fragment of the DBpedia page of Washington, Arkansas.

goal, all candidate entities, which can be referred by this surface form, are first selected out using the index established in stage 1. Then, some metrics need to be established to measure the likelihood that the surface form refers to an entity. Finally, the candidate entities whose likelihood values are larger than a threshold will be returned as the disambiguation result. The following sub sections will describe the metrics that can be used to measure this likelihood. We will also discuss the pros and cons of Wikipedia and DBpedia in deriving these metrics.

2.2.1 Entity Prominence

This metric is based on the relative importance of entities. For example, when the term *Washington* appears in a sentence, *Washington D.C.* generally has a higher prior probability to be the referred entity than another populated place also called *Washington*. In some existing works, such entity prominence are incorporated into the disambiguation model through hand crafted rules (e.g., countries are more important than cities) [19, 5].

In Wikipedia-based place name disambiguation, one can make use of the counts of *page-in* links (i.e. how many other Wikipedia articles have linked into this page) [8]. While the count values provide useful estimates, it is often time consuming to crawl the Wikipedia knowledge base to derive them. DBpedia, on the other hand, presents the counts of related links for free through the properties of *dbpedia-owl:wikiPageInLinkCount* and *dbpedia-owl:wikiPageOutLinkCount* (i.e., how many pages this article has linked out)[5]. These count values are derived when Wikipedia articles are converted into RDF, and therefore researchers can directly make use of these values instead of having to repeat the crawling process.

While page-in links are useful in quantifying the prominence of an entity, additional DBpedia properties, especially those describing the entity's geographic characteristics, can also be integrated to improve the disambiguation result. For example, when disambiguating populated places, we can employ their population values as additional information, and combine these values with the page-in counts. Equation 1 shows such an example metric.

$$P(s \rightarrow e_i) \propto \alpha \frac{Link(e_i)}{\sum_{j=1}^{n} Link(e_j)} + (1-\alpha)\frac{Popu(e_i)}{\sum_{j=1}^{n} Popu(e_j)} \quad (1)$$

where $P(s \rightarrow e_i)$ represents the probability that a surface form s refers to an entity e_i, $Link(e_i)$ represents the number of page-in links that entity e_i has, and $Popu(e_i)$ represents its population value; $\sum_{j=1}^{n} Link(e_j)$ sums up the number of links of all candidate entities e_j that the surface form s can refer to; $\alpha \in [0, 1]$ is a smoothing parameter which determines the relative importance of the two factors.

2.2.2 Context Similarity

Context has been considered as the *additional information which has impact on similarity judgment* [9], and has been examined in existing research [2, 1]. Context similarity is another metric used for named entity disambiguation. In existing works, such a metric has been performed using cooccurence models [17] as well as conceptual density [4]. In Wikipedia-based disambiguations, context similarity is often performed through vector space model which employs TF-IDF to assign weights to the vector terms [3, 6].

Compared with DBpedia, Wikipedia has been more frequently used to provide background information for candidate entities. Even *DBpedia Spotlight*, a notable named entity annotation system heavily based on DBpedia, also

[5] The link counts can also be downloaded from *Wikipedia Pagelinks* at http://wiki.dbpedia.org/Downloads39

utilizes Wikipedia articles for entity disambiguation [13]. One reason for the popularity of Wikipedia is due to its detailed descriptions, whereas DBpedia are more focused on representing entity relations. For example, a sentence like *"Greenville is one of the newest and smallest towns in Hillsborough County."*, would be reduced to the two RDF triples below (in Turtle syntax).

:Greenville a :Town.
:Greenville :isPartOf :Hillsborough_County.

While the skeleton information has been kept in the RDF triples, the descriptive terms, such as *"newest"* and *"smallest"*, are removed. In more extreme cases, sentences which do not involve any entities would be directly removed. For example, the following sentence, *"With a beautiful and historic downtown, Washington is known for the stately homes and lovely gardens that make up its residential area."*, will typically not survive the triplification process, although it conveys important descriptive information about the entity. However, the detailed descriptive information on Wikipedia also carries risks, since terms which represent important entities (e.g., a county name) may be assigned equal or even less weight as a purely descriptive word (e.g., *"beautiful"*).

2.2.3 Integrating Entity Prominence with Context Similarity

Both entity prominence and context similarity provide useful measures for the likelihood that a surface form may refer to a particular entity. Therefore, they have been integrated to improve the accuracy of named entity disambiguation. One such example is the work by Fader et al. that relies on a Bayesian approach [7]. Equation 2 illustrates the key idea:

$$e^* = \arg\max_{e_i \in E} [Sim(Context(s), Wiki(e_i)) \times P(s \to e_i)] \quad (2)$$

where s is the surface form, e_i is a candidate entity, and e^* represents the returned entity of the disambiguation; $Context(s)$ is the vector of context words surrounding s, and $Wiki(e_i)$ is the vector of Wikipedia article for entity e_i; $Sim(Context(s), Wiki(e_i))$ represents the cosine similarity between the two, and $P(s \to e_i)$ is the prior probability that s refers to e_i. Applying to a disambiguation task, this method will select the entities which have high prior probability to appear unless there is strong context evidence that suggests otherwise.

3. PROPOSED METHOD

The method proposed in this work focuses on the second stage of place name disambiguation. Thus, we assume that surface forms have been extracted from Wikipedia, and mapping between surface forms and their potential entities have been established using the procedure described in section 2.1. The research challenge we target in this paper is to disambiguate the candidate entities of a surface form given some context words. More specifically, we focus on the cases where the context is short, and therefore entity clues are important for the disambiguation task. The problem can be formalized as:

Problem. *Given a surface form s, its short context sentence $Context(s)$, and a set of candidate entities $E : \{e_1, e_2, ..., e_n\}$, return a subset E^* with entities that are more likely (larger than a threshold τ) to be referred to by s under the context.*

3.1 Enhancing TF-IDF Using DBpedia Terms

As discussed in the previous section, Wikipedia contains detailed descriptions about entities, but lacks the capability to differentiate terms representing entities from purely descriptive words. DBpedia, on the other hand, focuses on representing entities and their relations, but suffers from limited descriptive expressivity.

Based on this analysis, our first step is to combine the term frequency from both Wikipedia and DBpedia. Such a process reinforces the entity terms based on DBpedia while still keeping the descriptive words from Wikipedia. Reconsider the sentence mentioned before: *"Greenville is one of the newest and smallest towns in Hillsborough County."* While all the terms will be counted for the vector space model, the terms *town*, *Hillsborough*, and *County* will receive additional counts since they are mentioned in both Wikipedia and DBpedia. Please note that the phrase *Hillsborough County* has been divided into two terms in this step based on a unigram model. In addition, the frequency of *Greenville* is not considered since it is the surface form to be disambiguated.

Applying TF-IDF, we calculate the weight of a term t in the vector representation v_i of entity e_i using the following equations:

$$tf(t) = Freq_{wiki} + Freq_{dbpe} \quad (3)$$

$$idf(t) = 1 + \log(\frac{|E| + 1}{n_t}) \quad (4)$$

$$Weight(t) = tf(t) \times idf(t) \quad (5)$$

where $Freq_{wiki}$ and $Freq_{dbpe}$ are the frequency of term t from Wikipedia and DBpedia, respectively. $|E|$ represents the number of all candidate entities for surface form s, while n_t is the number of entities whose vectors contain the term t.

In typical natural language processing, numerical values are often removed in the pre-processing stage. In our case we deliberately keep the numerical values mentioned in DBpedia while discarding them in Wikipedia. This is because the numerical values in DBpedia often deliver important and unique information about the local place, such as population and total area. Thus, we incorporate these numbers into the vector space model as strings. However, this does not apply to Wikipedia, as it would introduce a lot of noise. For example, consider the following sentence from the Wikipedia page of *Washington, Arkansas*: *"Albert G. Simms (1882–1964), a United States Representative from New Mexico, was born here."* While the years are important information for describing the person, these numbers are not directly related to the place of *Washington, Arkansas*. DBpedia does not contain such facts (which are about a related person, not the place).

3.2 Integrating DBpedia Entities For Disambiguation

While we have partially integrated DBpedia facts into place name disambiguation, our method breaks the structured nature of DBpedia by dividing entity names into individual terms. In this section, we enhance the proposed method by integrating DBpedia entities into the disambiguation process.

The rationale behind this approach is based on the cooccurrence model from existing works [17, 12]. One geographic place is always associated with a unique set of other entities, such as nearby cities, higher-level administrative units, physical geographic features (e.g., rivers and moun-

Property	Associated entities
dbpedia-owl:country	Country
dbpedia-owl:isPartOf	State and county
dbpedia-owl:state	State
is dbpedia-owl:countySeat of	County
dbpprop:subdivisionName	Country, state, and county
is dbpedia-owl:location of	Buildings, parks, companies, or landmarks
is dbpedia-owl:city of	Schools and other organizations in that city
is dbpedia-owl:routeStart of	Routes (e.g., Highway 1) that starts from the place
is dbpedia-owl:routeEnd of	Routes that ended here
dbpedia-owl:district	The general district (e.g., dbpedia:St._Landry_Parish,_Louisiana)
dbpedia-owl:region	The general region
is dbpedia-owl:nearestCity of	The nearest city of this place
is dbpedia-owl:hometown of	People whose hometown is here
is dbpprop:birthPlace of	People who were born here
is dbpedia-owl:deathPlace of	People who passed away in this place
is dbpedia-owl:wikiPageRedirects of	Alias of the place
dbpprop:nickname	Nicknames

Table 1: DBpedia properties used to extract closely related entities.

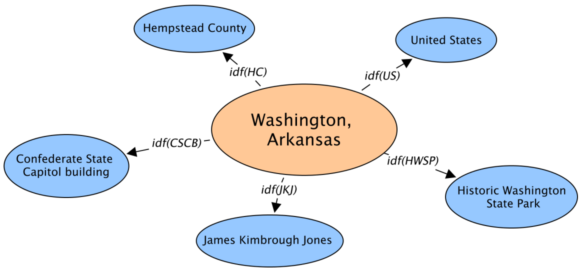


Figure 3: An entity graph of Washington, Arkansas.

tains), and related persons. The occurrence of a related entity can provide important clue for place name disambiguation. Specifically, we identify a list of DBpedia properties (Table 1) which associate one place to its closely related entities. Based on the selected DBpedia properties, a place can be considered as a central node within a simple graph. This central node links to other nodes (the associated entities), and the weights of the edges are determined using IDF as shown in equation 4. This way entities which are commonly associated with every candidate (e.g., United States) will be given a minimum weight, while entities which are unique to the local place will be assigned higher weights. Figure 3 shows such a simple graph for *Washington, Arkansas*, which links to other entities such as *Historic Washington State Park* and *James Kimbrough Jones*.

We then define an *entity matching score* in equation 6.

$$Match(Context(s), Entities(e_i)) = \frac{\sum_{j=1}^{m}(w_j \times I)}{\sum_{j=1}^{m} w_j} \tag{6}$$

where w_j is the weight for edge j; I is an indicator variable, and $I = 1$ when an edge match has been found in $Context(s)$; and $I = 0$ otherwise. $Entities(e_i)$ represents the entities closely related to e_i and extracted through the list of DBpedia properties in Table 1. Thus, given a sentence that contains a surface form s, the entity matching score is calculated by dividing the sum of the matched edge weights using the sum of all edge weights.

The entity matching score is then combined with the DBpedia-enhanced TF-IDF using a smoothing parameter λ.

$$\begin{aligned} S(s \to e_i) = &\lambda Match(Context(s), Entities(e_i)) + \\ &(1-\lambda)Sim(Context(s), WD(e_i))P(s \to e_i) \end{aligned} \tag{7}$$

where $\lambda \in [0, 1]$, and it controls the relative importance of the two parts in the equation. We will discuss how to adjust the value of λ in the experiments section. $WD(e_i)$ is the combined vector representation using the content from both Wikipedia and DBpedia. $Sim(Context(s), WD(e_i))$ is the cosine similarity between the context of the surface form s, and a candidate entity's vector representation; $P(s \to e_i)$ is the prior probability that s refers to e_i.

Given a short text, the cosine similarity value is often very small (e.g., in a scale of 10^{-5}) due to the small number of terms. Consequently, the calculated cosine similarity value will be on a smaller numeric scale compared to the entity matching score (which is often in the order of 10^{-1}). To ensure that the entity matching score does not dominate the result, a simple normalization function (equation 8) has been employed to convert the two scores to the same scale.

$$Normalized(x_i) = \frac{x_i - Min(x)}{Max(x) - Min(x)} \tag{8}$$

Based on the normalized values, the final score $S(s \to e_i)$ can be calculated using equation 7. We then define a sensitivity parameter τ. Candidate entities which have a score larger than τ will be returned as the disambiguation result, while the other candidates will be removed.

4. EXPERIMENTS

This section discusses experiments to evaluate the performance of the proposed method. We first describe the datasets used for the experiments, then present the results from applying our method and three other baseline methods to the datasets. Finally, we interpret the results and discuss their implications. All the experimental data and source code are available in our Github repository.

4.1 Datasets

The experimental datasets are derived based on a Wikipedia page which provides a list of the most common place names in the U.S.[6]. Based on this list, we selected two

[6] http://en.wikipedia.org/wiki/List_of_the_most_common_U.S.

place names *Washington* and *Greenville* as our experimental targets.

To acquire the ground truth data, we check the government websites of these cities and towns, and download the general descriptions of these places (which are often found in the *About* page). Such an approach reduces the amount of human interference in the ground truth data, compared with other existing methods (e.g., manually annotating the correct place names from texts). To evaluate the performance of the proposed method on short text, we designed a simple regular expression to break the long paragraphs of the descriptions into short sentences. While most of the ground truth data are in single sentences, there are also cases when two or three sentences are put together into one record. This happens because the downloaded descriptions occasionally do not follow the designed regular expression (e.g., *"."* may be used to finish the sentence instead of *". "*, with one more space after the period). In some other cases, one sentence is splitted in the middle due to the *". "* that are not used for period but for other purposes, such as *"Dr. "* or the name abbreviations *"Geo. H. Shaw"*. We deliberately include these special cases into our ground truth data, since the future data to be disambiguated may be noisy and may be poorly composed. Thus, we consider these special cases as a chance to test the robustness of the proposed method. The selected places are shown in table 2.

Washington	Greenville
Washington, Arkansas	Greenville, Alabama
Washington, Connecticut	Greenville, Georgia
Washington, Illinois	Greenville, Illinois
Washington, Iowa	Greenville, Indiana
Washington, Kansas	Greenville, Kentucky
Washington, Louisiana	Greenville, Mississippi
Washington, Maine	Greenville, North Carolina
Washington, New Jersey	Greenville, Pennsylvania
Washington, North Carolina	
Washington, Virginia	

Table 2: Experiment cities and towns.

Examples of two ground truth records are shown in table 3. The correct answer and the descriptive sentence are separated using a vertical bar (|).

Washington, New Jersey | Washington was also an important railroad center with multiple railroad stations and even a hotel across from one of the stations.
Greenville, Indiana | Early in Floyd County's history, Greenville was initially to be the county seat.

Table 3: Example records of the ground truth data.

We also download the text content from Wikipedia pages of those places as well as the structured data from the corresponding DBpedia page. Thus, the experimental datasets consist of three parts: the government ground truth descriptions, Wikipedia data, and DBpedia data. The average length of the test records in the Washington dataset is 27.8 words, while the average length for the Greenville dataset is 25.7 words (Note: stop words, such as *the* and *of*, also count towards the average length).

4.2 Experiment Procedure and Results

We use three baselines against which our method will be evaluated. All of them are based on the vector space model but with different data sources as background knowledge: using Wikipedia alone, using DBpedia alone, and using a combination of Wikipedia and DBpedia without entity matching scores. We implement these three methods, along with our proposed method. All of these implementation make use of both prior probability and context similarity as discussed in section 2.2.3. Different from the prior probability based on the counts of page-in links, our experiments derive prior probabilities based on the proportion of a city's ground truth records (see equation 9). This is because our experiments are in a controlled setting in which the prominence of an entity is not determined by the count of in-page links but by the number of sentences we can retrieve from their government websites (some cities have long descriptions while others have shorter texts). In real-world applications, prior probability calculation methods, such as equation 1, would be used.

$$P(s \rightarrow e_i) = \frac{N_i}{\sum_{j=1}^{n} N_j} \quad (9)$$

Here, $P(s \rightarrow e_i)$ is the prior probability that the surface form s refers to entity e_i; N_i is the number of descriptive records exist in the ground truth data for entity e_i, and $\sum_{j=1}^{n} N_j$ is the total number of records for all experimental cities and towns. To examine the randomness of the experimental data, we implement a program using only the prior probability, and run it 100 times on both the Washington dataset and the Greenville dataset. It obtains an average accuracy of 15.8% on Washington dataset, and 14.6% on the Greenville dataset.

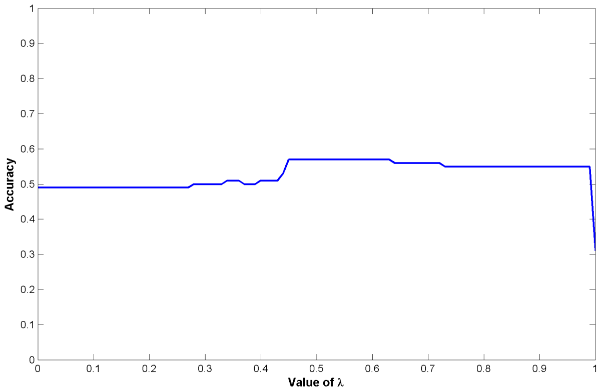


Figure 4: λ and accuracy plot for Washington experiment data.

To derive a proper value of the smoothing parameter λ for equation 7, we first analyze how it influences the disambiguation result. We iterate the value of λ from 0 to 1, and evaluate the quality of the first candidate (i.e., the candidate with the highest score) returned based on that λ value. We then calculate the accuracy value for each λ, and plot out the λ-accuracy figures for both of the two experimental cases (see Figure 4 and 5).

When λ equals to 0, the proposed method equals to the approach of only using a combined vector space model from Wikipedia and DBpedia (same as the baseline method 3). As λ increases, it enhances the impact of the entity matching score, while giving less importance to the context similarity and prior probability. Given a proper range of λ, the highest accuracy value is achieved. When λ approaches 1, the pro-

_place_names

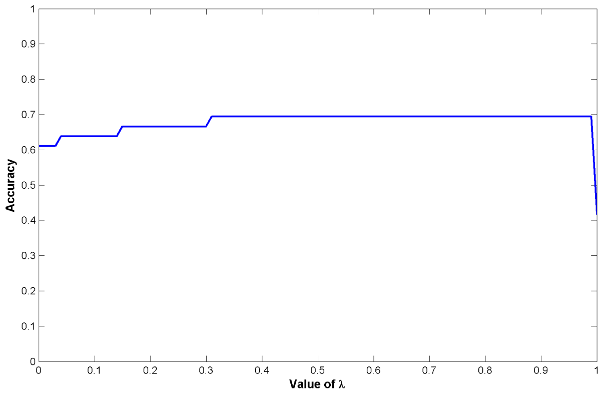


Figure 5: λ and accuracy plot for Greenville experiment data.

posed method is equal to using entity matching score alone, and the accuracy of the results drops dramatically.

As can be seen from the two figures, the accuracy only inceases mildly as λ iterates from 0 to 1. This can be attributed to three reasons. First, the improvement introduced by entity matching relies on the existence of entities in the descriptive sentences. However, there are sentences which are purely descriptive and do not contain any related entities. Second, while DBpedia is a comparatively complete knowledge database, it does not contain every entity related to a place. Thus, even when an entity exists in the sentence, such entity may not be identified by DBpedia. Finally, the improvement from entity matching is also based on the premise that using the combined vector space model alone will not disambiguate the place name correctly. Therefore, when the vector space model already generates correct result, including entity matching will not improve the accuracy (although the score of the correct candidate will be further increased).

Based on the above analysis, we select 0.5 for λ, since a balanced importance of the two components provides the best performance, as shown in Figure 4 and 5. We then applied the four methods to the two experimental datasets. The sensitivity parameter τ was iterated from 0 to 1, and the precisions and recalls of the four methods at each value of τ were calculated. We plotted the precision and recall curves for the two experimental datasets (Figure 6 and 7).

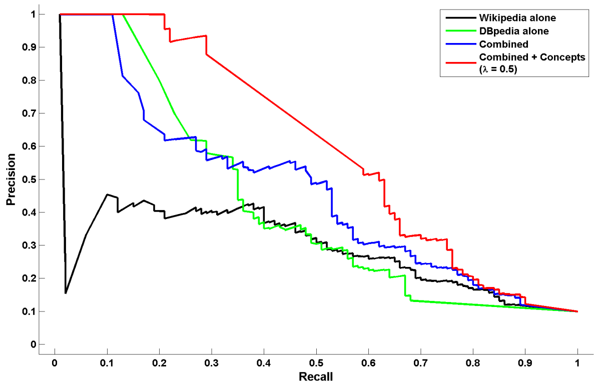


Figure 6: Precision and recall plot for Washington experiment data.

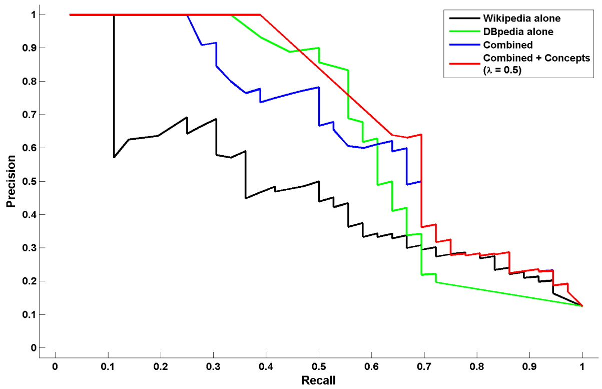


Figure 7: Precision and recall plot for Greenville experiment data.

4.3 Discussions

Precision and recall are a trade-off, and the curves in Figure 6 and 7 show the general performance of the four methods. In both experiments, the method of using Wikipedia alone achieves the lowest precision given a recall value between [0, 0.5]. Such a result confirms our analysis on the characteristics of Wikipedia data. While rich in descriptive sentences, Wikipedia data also contain sentences which are not directly about the places (e.g., stories about a person who lived there), thereby introducing noise to the background knowledge of the place.

On the contrary, using DBpedia alone results in high precision values if only low recall values are desired. This result is also consistent with our analysis, since DBpedia only provides information about the entities that are directly linked to the target place, thereby reducing the amount of noisy information. However, as the recall value increases, the precision of using DBpedia alone drops significantly, and becomes the lowest value among the four method for a recall larger than 0.7. This can be attributed to the lack of descriptive sentences in DBpedia, which limits the amount of background knowledge that can be used for place name disambiguation.

In both experiments, a combination of Wikipedia and DBpedia shows improvement over the previous two methods. In the low-recall region, the combined approach shows a precision which is much higher than the precision of using Wikipedia alone, although the precision is still lower than using DBpedia alone (since Wikipedia data also introduces noise). In the high-recall region, the combined approach provides a precision which is higher than using either Wikipedia or DBpedia alone.

Our proposed method further increases the performance of the combined approach. As can be seen from the two figures, given the highest precision, the proposed method effectively increases the recall without sacrificing precision. As the recall increases, the proposed method generally achieves the highest precision[7] compared with the three baselines in both of the datasets.

5. CONCLUSIONS AND FUTURE WORK

[7] There is a small fragment in Figure 7, where using DBpedia alone produces higher precision. Such a result can be attributed to the noise from Wikipedia data.

This paper proposed a method for improving Wikipedia-based place name disambiguation in short text using structured data from DBpedia. Wikipedia is a rich knowledge base which has often been employed to provide ground truth descriptions for places. However, the natural language representation of Wikipedia articles dilutes the weights of the important terms indicating related entities. Wikipedia also contains noise information which is not directly relevant to the target entities. DBpedia provides structured data about the properties of the target entity, as well as the relations between the target entity and other closely related entities. However, DBpedia lacks descriptive sentences which are also useful for place name disambiguation. Based on this analysis, we combine the merits of the two knowledge bases by adding up their term frequencies and incorporating an entity-matching mechanism. Experiments were performed to evaluate the proposed method against three baselines (using Wikipedia alone, using DBpedia alone, and using a combination of Wikipedia and DBpedia), and our method showed a better precision and recall balance.

This research can be further enhanced by some future work. So far, our experiments are based on datasets containing two place names, i.e., *Washington* and *Greenville*. While both place names are highly ambiguous, experiments on other datasets and places could help better evaluate the effectiveness of the proposed method. Another potential way to improve our method is to make use of the quantitative data (e.g., total area and population values) stored in DBpedia. As discussed in the paper, these numbers are often unique to local places. When given a sentence like, *"There are currently 15,134 people living in Washington, according to the 2010 census."*, the disambiguation engine should be able to recognize that this is about *Washington, Illionis* since the number of $15,134$ is unique to this Washington in terms of 2010 population. While our proposed method has kept these numbers, they are treated in the form of simple strings, and therefore are not given the importance as deserved. To better include them, the predicates of RDF triples, such as *dbpedia-owl:populationTotal*, should also be included in the disambiguation model.

6. REFERENCES

[1] M. Andrea Rodriguez and M. J. Egenhofer. Comparing geospatial entity classes: an asymmetric and context-dependent similarity measure. *International Journal of Geographical Information Science*, 18(3):229–256, 2004.

[2] M. Bazire and P. Brézillon. Understanding context before using it. In *Modeling and using context*, pages 29–40. Springer, 2005.

[3] R. C. Bunescu and M. Pasca. Using encyclopedic knowledge for named entity disambiguation. In *EACL*, volume 6, pages 9–16, 2006.

[4] D. Buscaldi and P. Rosso. A conceptual density-based approach for the disambiguation of toponyms. *International Journal of Geographical Information Science*, 22(3):301–313, 2008.

[5] D. Buscaldi, P. Rosso, and E. S. Arnal. *Using the wordnet ontology in the geoclef geographical information retrieval task*. Springer, 2006.

[6] S. Cucerzan. Large-scale named entity disambiguation based on wikipedia data. In *EMNLP-CoNLL*, volume 7, pages 708–716. Citeseer, 2007.

[7] A. Fader, S. Soderland, O. Etzioni, and T. Center. Scaling wikipedia-based named entity disambiguation to arbitrary web text. In *Proceedings of the IJCAI Workshop on User-contributed Knowledge and Artificial Intelligence: An Evolving Synergy, Pasadena, CA, USA*, pages 21–26, 2009.

[8] J. Hoffart, M. A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum. Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 782–792. Association for Computational Linguistics, 2011.

[9] K. Janowicz. Kinds of contexts and their impact on semantic similarity measurement. In *Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on*, pages 441–446. IEEE, 2008.

[10] C. B. Jones and R. S. Purves. Geographical information retrieval. *International Journal of Geographical Information Science*, 22(3):219–228, 2008.

[11] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 2014.

[12] J. L. Leidner. *Toponym resolution in text: Annotation, evaluation and applications of spatial grounding of place names*. Universal-Publishers, 2008.

[13] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 1–8. ACM, 2011.

[14] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242. ACM, 2007.

[15] D. Milne and I. H. Witten. Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 509–518. ACM, 2008.

[16] H. T. Nguyen and T. H. Cao. Named entity disambiguation on an ontology enriched by wikipedia. In *Research, Innovation and Vision for the Future, 2008. RIVF 2008. IEEE International Conference on*, pages 247–254. IEEE, 2008.

[17] S. Overell and S. Rüger. Using co-occurrence models for placename disambiguation. *International Journal of Geographical Information Science*, 22(3):265–287, 2008.

[18] D. A. Smith and G. Crane. Disambiguating geographic names in a historical digital library. In *Research and Advanced Technology for Digital Libraries*, pages 127–136. Springer, 2001.

[19] R. Volz, J. Kleb, and W. Mueller. Towards ontology-based disambiguation of geographical identifiers. In *I3*, 2007.