

# Geographic Named Entity Disambiguation with Automatic Profile Generation

Yefei Peng, Daqing He, and Ming Mao  
School of Information Sciences  
University of Pittsburgh  
Pittsburgh, PA 15206 USA  
{ypeng,daqing, mingmao}@mail.sis.pitt.edu

## Abstract

*Knowledge rich approach of processing documents has been viewed as a method to improve over simple bag-of-word representation. Extracting location information from documents and link them to some ontology such as world gazetteer through a disambiguation process becomes an interesting and important topic. Lacking of training data is a problem in disambiguation method. In this paper we described a method to automatically extract training data from large collection of documents based on local context disambiguation, and then sense profiles are generated automatically for disambiguation use. Another topic of this paper is to describe a linear combination method to combine different types of evidences of disambiguation. We explored three different evidences including location sense context in training documents, local neighbor context, and the popularity of individual location sense. Our results show that combining the three evidences generates reasonable results.*

## 1. Introduction

There are two concerns in the core of information science, first one is how can we understand user's quest, second one is how can we represent knowledge structures. These two concerns depend on each other. The more we understand user's need, the better we can represent knowledge structure, and vice versa.

For the second concern, how to represent knowledge structure, ontology-based knowledge representation is a widely used method. But currently, there is not enough ontology to use. How to extract knowledge from text is another big problem. A compromise method is: instead of trying to fully understand the meaning of text, we can simply extract named entities from text. Then some data mining system could be used to extract knowledge from named entities.

Location information is an important piece of information in text. It constrains the context of the

documents to certain geographic areas, and builds up a direct link between the descriptions presented in the documents and those locations. In addition, there are many different versions of gazetteers available online, which can be used as the ontology for differentiating locations and expressing the relationship among those locations. Therefore, if we can extract locations from text, and connect them to gazetteer, the information process system would have a clear picture about which locations that the documents are talking about. Locations, and persons, organizations are often called named entities in literature [1].

Just like human can perform many different actions once they know more about a topic, information processing system can develop many ways to explore the usages of location information in documents once such knowledge is available. For example, some geographic scope constraint can be imposed on the queries to these documents. Search results could be connected to gazetteer and be visualized to show the geographic distribution. Relation between locations and other type of extracted information could be analyzed, utilized, and presented. We can also connect user modeling with location information to explore, for example, adaptive news services based on user's current location.

However, identifying a correct node on a gazetteer for a given location term usually is not straightforward because some locations share the same name, whereas the same location might be called differently in different context and time period. The fact that people often use abbreviations in referring locations adds the complexity of this problem. For example, there are 22 cities and 1 state named "Washington" in the United States alone.

In this paper, we examined the extraction of terms related to locations, and the disambiguation process of identifying correct ontology nodes for these location terms. Specifically, we proposed an approach to automatically get training data for disambiguation, which avoids manual annotations. Then we used these data to build profile for each sense. These profiles could be used in disambiguation. Besides profiles, several other

resources have been explored during disambiguation process in the literature. This includes, local context information, default meaning, single sense per discourse, preferring most frequent meaning, etc. We also include these types of information in our disambiguation approach.

## 2. Related Work

Recently the task of location disambiguation has been gaining attention. Apparently this task requires general knowledge of locations in the world. World gazetteer listing all names of places provides this general knowledge. Most published work relied on various natural language processing heuristics.

The rigorous step-by-step algorithm presented by Li *et al.* [2, 3] is a typical method. First, look up location names in the gazetteer to associate candidate sense for each location mention. Next, NLP techniques are called upon to help weed out non-geo terms. Then, the “single sense per discourse” principle is applied. Next, a graph search algorithm is applied to resolve remaining ambiguities. Then names that remain unresolved are assigned a default sense, the most important one associated with the given name.

The algorithm presented by Leidner *et al.* [4] is similar to previous method. Bilhaut *et al.*[5] expanded the language understanding step, so that phrases such as “north of France” could be interpreted. Amitay *et al.* [6] not only did disambiguation, but also assigned each web page a geographic focus, which is a locality that the page discusses as a whole.

These previous related work normally employed hybrid algorithm to combine evidences and set of heuristic rules. There is not a formal way to combine the evidences from every step. They also did not employ machine learning, because training data is not easy to get. It requires many manual annotations which is time consuming.

## 3. Approach

Context information plays major role in our approach. There are two types of context: local context and global context. Local context of a location mention is direct neighbor words of the mention in a document. For example, if “Aberdeen” is a location mention and followed by “Scotland”, then “Scotland” is local context of “Aberdeen”. Global context of a sense is frequently co-located words of the sense in a collection of documents. For example, “Washington, D.C” may have global context of “President Bush, “White House””; while

“Washington state” may have context of “west”, “Seattle”.

We believe that every sense of location entity has different global context. If we could build a profile for each sense, in future, we can disambiguate location entity by looking for similar context in profiles. But the difficult part is how to get training data for profiles building. Manual annotation is one way to do it, but it is time consuming and not appropriate for large collection of data. So our thinking is if we can automatically disambiguate some location mentions with high confidence, they can be used as training data. Even though they could be a small portion of all mentions, considering we have enough large collection of documents and the process is automatic, we can still get enough training data. Now the question is how to automatically disambiguate even a small portion of mentions? The answer is to use local context.

Context of a mention of location can directly help disambiguation. Because writer also knows there will be disambiguation, so he may use other indicator to uniquely or more specifically disambiguate the mention of location. For example, “Aberdeen, Maryland” and “Aberdeen, Scotland” can be easily disambiguated. “Maryland” and “Scotland” are local direct context of the two “Aberdeen” respectively. If we can disambiguate a mention with this type of local context, we are pretty sure it is correct.

In our approach, only the word before and the word after the location mention is extracted, if the parent/child/country node of possible sense in ontology appears in this local context, this qualified sense will be assigned to the mention with confidence score of 1. We can process a large collection of documents with this automatic method and get enough training data.

After getting training data, we extract context of every location sense in training data, so that different sense gets different global context. For each sense, we generate a profile which contains all context of the sense within the collection. All profiles are index by Indri search engine v2.2 [7]. When an entity needs to be disambiguated, its local context will be extracted as query to Indri; the top 10 returned senses will get a confidence score for each. The first returned sense will get a confidence score of 1; the second will get a confidence score of 0.9, and so on. We call this score  $S_i$ .

If local context exists for a mention, then we absolutely should use it. The word before and the word after the location mention is extracted, if the parent/child/country node of possible sense in ontology appears in this local context, this qualified sense will be assigned to the mention with confidence score of 1. We call this score  $S_c$ .

Possible senses of a location mention have different probability. If a sense is popular, the probability is high. Here we use population to represent the popularity of a sense. If there are  $N$  senses with the same name, they will be ranked by population. The sense with largest popularity will be given a confidence score of 1. The  $n$ th sense in the ranked list will get a confidence score of  $(N-n+1)/N$ . We call this score  $S_p$ .

Final score will be calculated as weighted sum of all the three scores:

$$S_{final} = w_t S_t + w_c S_c + w_p S_p \quad (1)$$

Then the sense with largest score will be assigned to the location entity.

## 4. Experiment

In our experiment, we used a named entity extraction tool developed by IBM. It can extract location named entity, and co-reference information about the locations within document. The world gazetteer was downloaded from the Web [8] and expanded. It has 171,021 nodes, 5 levels, and concentrates on world. Besides the location information, it also has population information, and some information such as ordinary alternative names was added.

Our collection of documents is a portion of TDT4 collection [9]. They are English newspaper articles from major news agencies in US, including New York Times, Associated Press, and Voice of America. Totally there are 17,755 documents. The average length of these articles is 2,638 bytes.

Named entity extraction tool was run on this collection, extracted location mentions were saved in to database for later use. Then local context of each mention was examined, if the parent/child/country node of possible sense in ontology appears in this local context, this qualified sense will be assigned to the mention with confidence score of 1. These recognized mentions were treated as training data. Based on these training data, global profiles for each sense were constructed. For each sense, all contexts of the sense were extracted and put in to one file. This file is the global profile of the sense. Then all profiles were indexed by Indri 2.2 search engine for later use.

To test the accuracy of our approach, we randomly selected 300 English newspaper articles from this collection as test collection, and manually constructed ground truth for the location information for the experiment purpose. After running named entity extraction tool, every location instance in text was manually disambiguated, that is, connected to an entity in the gazetteer. Totally there are 2220 location entities in testing collection.

Disambiguation based on local context as described above was tested on the testing collection. The accuracy is 100%. This confirmed our assumption that local context disambiguation has very high accuracy. In the whole collection, there are 4498 out of 168, 341 location mentions were disambiguated by local context. They were used as training data in global context generation.

In our result, location mentions will be divided into five categories:

- A. This is a location, and our algorithm is correct.
- B. This is a location, but our algorithm does not rank the correct sense as number 1.
- C. This is a location, but our algorithm classifies it as non location.
- D. This is not a location, and our algorithm is correct.
- E. This is not a location, but our algorithm classifies it as a location.

They are also shown in Table 1.

**Table 1 Result categorization**

Classified result	Ground truth	
	Location	Non-location
Correct	A	D
Wrong	B (not first)	E
	C (non-location)	

We define accuracy as:

$$\text{Accuracy} = \frac{A + D}{A + B + C + D + E}$$

The default setting of parameters in Equation (1) is: all weights are one. Global context window size is four words before and five words after location mention. All weights ( $w_c$ ,  $w_p$  and  $w_t$ ) are set to 1. The accuracy is 76.4%. In Table 2, category distribution of location entities are listed.

**Table 2. Category distribution of location entities**

Category	A	B	C	D	E
# of Entities	731	230	127	964	168

Then we adjust the weights as follows. Since the accuracy of local context disambiguation is high, we set  $w_c$  as 2.  $w_p$  is not touched.  $w_t$  is set to 0.5. The accuracy is 78.8%. In Table 3, category distribution of location entities are listed.

**Table 3. Category distribution of location entities**

Category	A	B	C	D	E
# of Entities	787	181	127	964	168

If we compare Table 2 and Table 3, we can see that the improvement in accuracy came from category B. Part of category B in Table 2 goes to category A in Table 3. It means, in Table 2, the correct sense of some entities are not ranked top 1, but in Table 3, they are ranked top 1.

Then we compare the accuracy for different context length. The results are shown in Table 4. We can see that optimal length is 4 words before and 4 words after location mentions.

**Table 4. Accuracy with different context length**

$L$ (word #)	Accuracy (%)
1	75.4
2	76.2
3	78.4
4	78.8
5	76.7

## 5. Discussion

How to automatically get training data for location disambiguation is a question. Manual annotation is time and money consuming. In our work, we showed that automatically extracting training data from large collection by local context disambiguation is a promising method.

How to determine the right context size is a question. Too many contexts would introduce noise; require more computational power, and larger index size. Too few contexts will end up with inadequate contextual information; therefore the recall value might drop. From results shown in Table 3, we can conclude that including four words before and four words after the location mention may be a good choice.

Another question about evidence combination is the optimal weights for each type of evidence. For example, should the weight for local context be more important than that of popularity? Experiment results in Table 2 and 3 show that local context is more important than other types of evidence.

Based on data in Table 2 and 3, we examine further on the origins of the errors. For category C where location entities are wrongly classified as non-location, we found that most of the errors are because the corresponding entities were not available in the gazetteer. For example, terms referring to geographic areas, like Middle East, West Bank, are not in the ontology. Therefore, no matter how we adjust our algorithms, we can not make them right. Another set of errors are from the different judgments on what constitutes a location. For example, the IBM named entity tagger marked up "road" and "home" as locations, whereas our annotators did not think these are real locations, and there are no corresponding nodes in gazetteer for them. Category B is where our algorithm could be improved. In category B, location entities are classified as location entities, but correct sense is not ranked top 1. Better disambiguation

process can be developed to move those instances up to the top rank.

## 6. Conclusion

In this paper, we have described an approach to automatically extract location disambiguation training data from large collection. These data could be used to build sense profile for later disambiguation. We also explore combination evidence from multiple sources in location entity disambiguation. We examined the way to extract global context information based on collocation information, and local specific context information that is related the specific mention of the location entity. Our combination uses a linear combination approach with different weights. Our experiments indicate that different weights should be applied to different sources.

Our future work includes investigation of different term weighting methods for sense profile, e.g. tf-idf, mutual information, etc. To prevent errors in category C, we should expand our gazetteer to include areas, e.g., Middle East, West Bank, etc.

## 7. References

- [1] "[http://www.cs.nyu.edu/cs/faculty/grishman/NETask20.book\\_1.html](http://www.cs.nyu.edu/cs/faculty/grishman/NETask20.book_1.html)."
- [2] H. Li, R. Srihari, C. Niu, and W. Li, "Location normalization for information extraction," in *Proc. of the 19th Conference on Computational Linguistics*. Taipei, Taiwan, 2002.
- [3] H. Li, R. K. Srihari, C. Niu, and W. Li, "InfoXtract location normalization: a hybrid approach to geographic," in *Workshop on the Analysis of Geographic References*. Edmonton, Canada, 2003.
- [4] J. Leidner, G. Sinclair, and B. Webber, "Grounding spatial named entities for information extraction and question answering," in *Workshop on the Analysis of Geographic References*. Edmonton, Alberta, Canada, 2003.
- [5] F. Bilhaut, T. Charois, P. Enjalbert, and Y. Mathet, "Geographic Reference Analysis for Geographic Document Querying," in *Workshop on the Analysis of Geographic References*. Edmonton, Alberta, Canada, 2003.
- [6] E. Amitay, N. Har'El, R. Sivan, and A. Soffer, "Web-where: geotagging web content," presented at SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, Sheffield, United Kingdom, 2004.
- [7] "<http://www.lemurproject.org/indri/>."
- [8] "<http://www.world-gazetteer.com/>."
- [9] "<http://projects.ldc.upenn.edu/TDT4/>."