Exploring Entity Relations for Named Entity Disambiguation

Danuta Ploch
DAI-Labor, Technische Universität Berlin
Berlin, Germany
danuta.ploch@dai-labor.de

Abstract

Named entity disambiguation is the task of linking an entity mention in a text to the correct real-world referent predefined in a knowledge base, and is a crucial subtask in many areas like information retrieval or topic detection and tracking. Named entity disambiguation is challenging because entity mentions can be ambiguous and an entity can be referenced by different surface forms. We present an approach that exploits Wikipedia relations between entities co-occurring with the ambiguous form to derive a range of novel features for classifying candidate referents. We find that our features improve disambiguation results significantly over a strong popularity baseline, and are especially suitable for recognizing entities not contained in the knowledge base. Our system achieves state-of-the-art results on the TAC-KBP 2009 dataset.

1 Introduction

Identifying the correct real-world referents of named entities (NE) mentioned in text (such as people, organizations, and geographic locations) plays an important role in various natural language processing and information retrieval tasks. The goal of Named Entity Disambiguation (NED) is to label a surface form denoting an NE in text with one of multiple predefined NEs from a knowledge base (KB), or to detect that the surface form refers to an out-of-KB entity, which is known as NIL detection. NED has become a popular research field recently, as the growth of large-scale publicly available encyclopedic knowledge resources such as Wikipedia has stimulated research on linking NEs in text to their entries in these KBs (Bunescu and Pasca, 2006; McNamee and Dang, 2009).

The disambiguation of named entities raises several challenges: Surface forms in text can be ambiguous, and the same entity can be referred to by different surface forms. For example, the surface form "George Bush" may denote either of two former U.S. presidents, and the later president can be referred to by "George W. Bush" or with his nickname "Dubya". Thus, a many-to-many mapping between surface forms and entities has to be resolved. In addition, entity mentions may not have a matching entity in the KB, which is often the case for non-popular entities.

Typical approaches to NED combine the use of document context knowledge with entity information stored in the KB in order to disambiguate entities. Many systems represent document context and KB information as word or concept vectors, and rank entities using vector space similarity metrics (Cucerzan, 2007). Other authors employ supervised machine learning algorithms to classify or rank candidate entities (Bunescu and Pasca, 2006; Zhang et al., 2010). Common features include popularity metrics based on Wikipedia's graph structure or on name mention frequency (Dredze et al., 2010; Han and Zhao, 2009), similarity metrics exploring Wikipedia's concept relations (Han and Zhao, 2009), and string similarity features. Recent work also addresses the task of NIL detection (Dredze et al., 2010).

While previous research has largely focused on disambiguating each entity mention in a document

Proceedings of the ACL-HLT 2011 Student Session, pages 18–23,
Portland, OR, USA 19-24 June 2011.

separately (McNamee and Dang, 2009), we explore an approach that is driven by the observation that entities normally co-occur in texts. Documents often discuss several different entities related to each other, e.g. a news article may report on a meeting of political leaders from different countries. Analogously, entries in a KB such as Wikipedia are linked to other, related entries.

Our Contributions In this paper, we evaluate a range of novel disambiguation features that exploit the relations between NEs identified in a document and in the KB. Our goal is to explore the usefulness of Wikipedia's link structure as source of relations between entities. We propose a method for candidate selection that is based on an inverted index of surface forms and entities (Section 3.2). Instead of a bag-of-words approach we use co-occurring NEs in text for describing an ambiguous surface form. We introduce several different disambiguation features that exploit the relations between entities derived from the graph structure of Wikipedia (Section 3.3). Finally, we combine our disambiguation features and achieve state-of-the-art results with a Support Vector Machine (SVM) classifier (Section 4).

2 Problem statement

The task of NED is to assign a surface form s found in a document d to a target NE $t \in E(s)$, where $E(s) \subset E$ is a set of candidate NEs from an entity KB that is defined by $E = \{e_1, e_2, ..., e_n\}$, or to recognize that the found surface form s refers to a missing target entity $t \notin E(s)$. For solving the task, three main challenges have to be addressed:

Ambiguity Names of NEs may be ambiguous. Since the same surface form s may refer to more than one NE e, the correct target entity t has to be determined from a set of candidates $E(s)$

Name variants Often, name variants (e.g. abbreviations, acronyms or synonyms) are used in texts to refer to the same NE, which has to be considered for the determination of candidates $E(s)$ for a given surface form s.

KB coverage KBs cover only a limited number of NEs, mostly popular NEs. Another challenge of NED is therefore to recognize missing NEs where $t \notin E(s)$, given a surface form s (NIL detection).

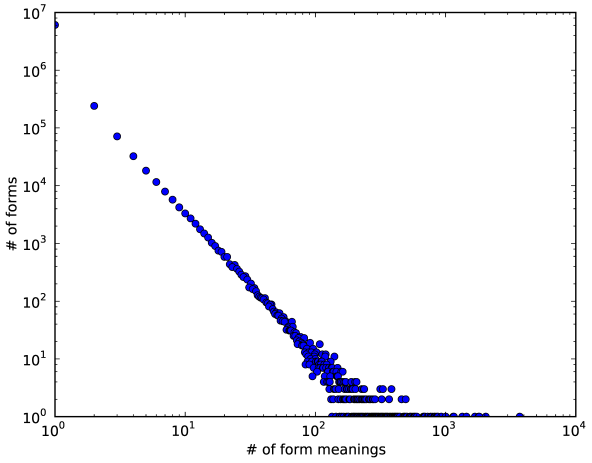

Figure 1: Ambiguity of Wikipedia surface forms. The distribution follows a power law, as many surface forms have only a single meaning (i.e. refer to a single Wikipedia concept), and some surface forms are highly ambiguous, referring to very many different concepts.

3 Named Entity Disambiguation

We formulate NED as a supervised binary classification problem. In this section we describe the construction and structure of the KB and the candidate selection scheme, followed by an overview of disambiguation features and the candidate classification algorithm.

3.1 Knowledge base construction

Our approach disambiguates named entities against a KB constructed from Wikipedia. To this end, we process Wikipedia to extract several types of information for each Wikipedia article describing a concept (i.e. any article not being a redirect page, a disambiguation page, or any other kind of meta page). We collect a set of name variants (surface forms) for each concept from article titles, redirect pages, disambiguation pages and the anchor texts of internal Wikipedia links, following Cucerzan (2007). For each concept, we also collect its set of incoming and outgoing links to other Wikipedia pages. Finally, we extract the article's full text. We store this information in an inverted index, which allows for very efficient access and search during candidate selection and feature computation.

The distribution of surface forms follows a power law, where the majority of surface forms is unambiguous, but some surface forms are very ambiguous (Figure 1). This suggests that for a given set of distinct surface forms found in a document, many of these will unambiguously refer to a single Wikipedia entity. These entities can then be used to disambiguate surface forms referring to multiple entities.

3.2 Candidate selection

Given a surface form identified in a document, the task of the candidate selection component is to retrieve a set of candidate entities from the KB. To this end, we execute a search on index fields storing article titles, redirect titles, and name variants. We implement a weighted search to give high weights to exact title matches, a lesser emphasis on redirect matches, and finally a low weight for all other name variants. In addition, we implement a fuzzy search on the title and redirect fields to select KB entries with approximate string similarity to the surface form.

3.3 Disambiguation features

In this section, we describe the features that we use in our disambiguation approach.

Entity Context (EC) The EC disambiguation feature is calculated as the cosine similarity between the document context **d** of a surface form s and the Wikipedia article **c** of each candidate $c \in E(s)$. We represent both contexts as vectors of URIs. To create **d** we extract all NEs from the text using the Stanford NE Recognizer (Finkel et al., 2005) and represent each NE by its Wikipedia URI. If a surface form is ambiguous, we choose the most popular NE with the popularity metric described below. Analogously, we represent each c as a vector of the incoming and outgoing URIs found on its Wikipedia page.

Link Context (LC) The link context feature is an extension of the EC feature. Since our observations have shown that the entity context can be very small and consequently the overlap between **d** and **c** may be very low, we extend **d** by all incoming (LC-in) or by all incoming and outgoing (LC-all) Wikipedia URIs of the NEs from the entity context. We assume that Wikipedia pages that refer to other Wikipedia pages contain information on the referenced pages or at least are thematically related to these pages. With the extension of **d** to **d'**, we expect a higher overlap between the context vectors, so that $\cos(\mathbf{d'}, \mathbf{c}) \geq \cos(\mathbf{d}, \mathbf{c})$.

Candidate Rank (CR) The features described so far disambiguate every surface form $s \in S$ from a document d separately, whereas our Candidate Rank feature aims to disambiguate all surface forms S found in a document d at once. We represent d as a graph $D = (E(S), L(E(S)))$ where the nodes $E(S) = \cup_{s \in S} E(s)$ are all candidates of all surface forms in the document and $L(E(S))$ is the set of links between the candidates, as found in Wikipedia. Then, we compute the PageRank score (Brin and Page, 1998) of all $c \in E(S)$ and choose for each s the candidate with the highest PageRank score in the document graph D.

Standard Features In addition to the previously described features we also implement a set of commonly accepted features. These include a feature based on the cosine similarity between word vector representations of the document and the Wikipedia article of each candidate (BOW) (Bunescu, 2007). We perform stemming, remove stopwords, and weight words with tf.idf in both cases. Another standard feature we use is the popularity of a surface form (SFP). We calculate how often a surface form s references a candidate $c \in E(s)$ in relation to the total number of mentions of s in Wikipedia (Han and Zhao, 2009). Since we use an index for selecting candidates (Section 3.2), we also exploit the candidate selection score (CS) returned for each candidate as a disambiguation feature.

3.4 Candidate classifier and NIL detection

We cast NED as a supervised classification task and use two binary SVM classifiers (Vapnik, 1995). The first classifier decides for each candidate $c \in E(s)$ if it corresponds to the target entity. Each candidate is represented as a vector $\mathbf{x}^{(c)}$ of features. For training the classifier we label as a positive example at most one $\mathbf{x}^{(c)}$ from the set of candidates for a surface form s, and all others as negative.

In addition, we train a separate classifier to detect NIL queries, i.e. where all $\mathbf{x}^{(c)}$ from $E(s)$ are labeled as negative examples. This may e.g. be the case

	All queries	KB	NIL
Baseline features	0.7797	0.6246	0.8964
All features	0.8391	0.6795	0.9592
Best features	0.8422	0.6825	0.9623
Dredze et al.	0.7941	0.6639	0.8919
Zheng et al.	0.8494	0.7900	0.8941
Best TAC 2009	0.8217	0.7725	0.8919
Median TAC 2009	0.7108	0.6352	0.7891

Table 1: Micro-averaged accuracy for TAC-KBP 2009 data compared for different feature sets. The best feature set contains all features except for LC-all and CR. Our system outperforms previously reported results on NIL queries, and compares favorably on all queries.

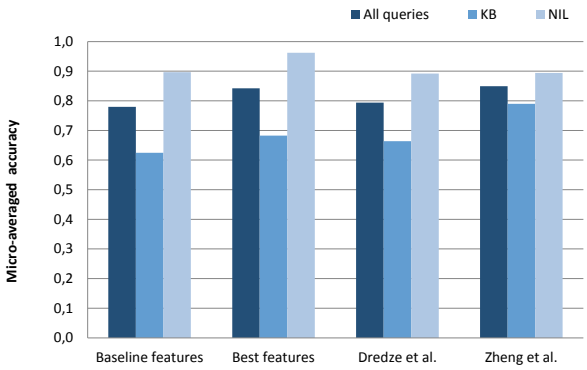

Figure 2: The micro-averaged accuracy for all types of queries on TAC-KBP 2009 data in comparison to other systems.

if the similarity values of all candidates $c \in E(s)$ are very low. We calculate several different features, such as the maximum, mean and minimum, the difference between maximum and mean, and the difference between maximum and minimum, of all atomic features, using the feature vectors of all candidates in $E(s)$. Both classifier use a radial basis function kernel, with parameter settings of $C = 32$ and $\gamma = 8$. We optimized these settings on a separate development dataset.

4 Evaluation

We conduct our experiments on the 2009 Knowledge Base Population (KBP) dataset of the Text Analysis Conference (TAC) (McNamee and Dang, 2009). The dataset consists of a KB derived from a 2008 snapshot of the English Wikipedia, and a collection of newswire, weblog and newsgroup documents. A set of 3904 surface form-document pairs (queries) is constructed from these sources, encompassing 560 unique entities. The majority of queries (57%) are NIL queries, of the KB queries, 69% are for organizations and 15% each for persons and geopolitical entities. For each query the surface form appearing in the given document has to be disambiguated against the KB.

We randomly split the 3904 queries to perform 10-fold cross-validation, and stratify the resulting folds to ensure a similar distribution of KB and NIL queries in our training data. After normalizing feature values to be in $[0, 1]$, we train a candidate and a NIL classifier on 90% of the queries in each iteration, and test using the remaining 10%. Results reported in this paper are then averaged across the test folds.

Table 1 compares the micro-averaged accuracy of our approach on KB and NIL queries for different feature sets, and lists the results of two other state-of-the-art systems (Dredze et al., 2010; Zheng et al., 2010), as well as the best and median reported performance of the 2009 TAC-KBP track (McNamee et al., 2010). Micro-averaged accuracy is calculated as the fraction of correct queries, and is the official TAC-KBP evaluation measure. As a baseline we use a feature set consisting of the BOW and SFP features. The best feature set in our experiments comprises all features except for the LC-all and CR features.

Our best accuracy of 0.84 compares favorably with other state-of-the-art systems on this dataset. Using the best feature set improves the disambiguation accuracy by 6.2% over the baseline feature set, which is significant at $p = 0.05$. For KB queries our system's accuracy is higher than that of Dredze et al., but lower than the accuracy reported by Zheng et al. One striking result is the high accuracy for NIL queries, where our approach outperforms all previously reported results (Figure 2).

Figure 3 displays the performance of our approach when iteratively adding features. We can see that the novel entity features contribute to a higher overall accuracy. Including the candidate selection score (CS) improves accuracy by 3.6% over the baseline. The Wikipedia link-based features provide additional gains, however differences are quite

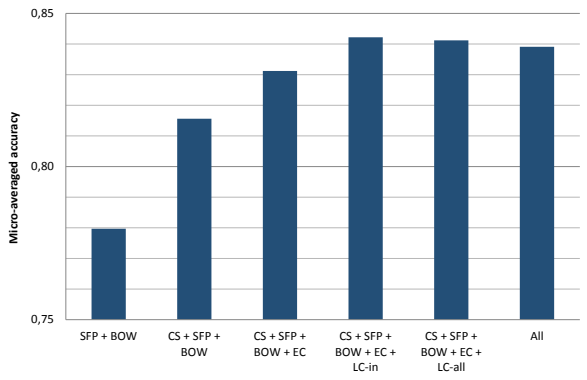

Figure 3: Differences in micro-averaged accuracy for various feature combinations on TAC-KBP 2009 data. Adding Wikipedia link-based features significantly improves performance over the baseline feature set.

small ($1.0 - 1.5\%$). We find that there is hardly any difference in performance between using the LC-all and LC-in features. The Candidate Rank (CR) feature slightly decreases the overall accuracy. A manual inspection of the CR feature shows that often candidates cannot be distinguished by the classifier because they are assigned the same PageRank scores. We assume this results from our use of uniform priors for the edges and vertices of the document graphs.

5 Conclusion and Future Work

We presented a supervised approach for named entity disambiguation that explores novel features based on Wikipedia's link structure. These features use NEs co-occurring with an ambiguous surface form in a document and their Wikipedia relations to score the candidates. Our system achieves state-of-the-art results on the TAC-KBP 2009 dataset. We find that our features improve disambiguation results by 6.2% over the popularity baseline, and are especially helpful for recognizing entities not contained in the KB.

In future work we plan to explore multilingual data for NED. Since non-English versions of Wikipedia often are less extensive than the English version we find it promising to combine Wikipedia versions of different languages and to use them as a source for multilingual NED. For multilingual NED evaluation we are currently working on a German dataset, following the TAC-KBP dataset creation guidelines. In addition to Wikipedia, we also intend to exploit more dynamical information sources. For example, when considering news articles, NEs often occur for a certain period of time in consecutive news dealing with the same topic. This short-time context could be a useful source of information for disambiguating novel entities.

References

Sergey Brin and Lawrence Page. 1998. The anatomy of a large-scale hypertextual web search engine. In *WWW7: Proceedings of the seventh international conference on World Wide Web 7*, pages 107–117, Amsterdam, The Netherlands. Elsevier Science Publishers B. V.

Razvan Bunescu and Marius Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06)*, pages 9–16, Trento, Italy.

Razvan Constantin Bunescu. 2007. *Learning for Information Extraction: From Named Entity Recognition and Disambiguation To Relation Extraction*. Ph.D. thesis, University of Texas at Austin, Department of Computer Sciences.

Silviu Cucerzan. 2007. Large-Scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 708–716, Prague, Czech Republic. Association for Computational Linguistics.

Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. 2010. Entity disambiguation for knowledge base population. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 277–285, Beijing, China. Coling 2010 Organizing Committee.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370, Ann Arbor, Michigan. Association for Computational Linguistics.

Xianpei Han and Jun Zhao. 2009. Named entity disambiguation by leveraging wikipedia semantic knowledge. In *Proceeding of the 18th ACM conference on Information and knowledge management*, pages 215–224, Hong Kong, China. ACM.

Paul McNamee and Hoa Trang Dang. 2009. Overview of the tac 2009 knowledge base population track. In *Text Analysis Conference (TAC)*.

Paul McNamee, Hoa Trang Dang, Heather Simpson, Patrick Schone, and Stephanie M. Strassel. 2010. An evaluation of technologies for knowledge base population. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. European Language Resources Association (ELRA).

Vladimir N. Vapnik. 1995. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Wei Zhang, Jian Su, Chew Lim Tan, and Wen Ting Wang. 2010. Entity linking leveraging automatically generated annotation. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1290–1298, Beijing, China. Coling 2010 Organizing Committee.

Zhicheng Zheng, Fangtao Li, Minlie Huang, and Xiaoyan Zhu. 2010. Learning to link entities with knowledge base. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 483–491, Stroudsburg, PA, USA. Association for Computational Linguistics.