Local and Global Algorithms for Disambiguation to Wikipedia

Lev Ratinov [1] **Dan Roth**[1] **Doug Downey**[2] **Mike Anderson**[3]

[1]University of Illinois at Urbana-Champaign
{ratinov2|danr}@uiuc.edu

[2]Northwestern University
ddowney@eecs.northwestern.edu

[3]Rexonomy
mrander@gmail.com

Abstract

Disambiguating concepts and entities in a context sensitive way is a fundamental problem in natural language processing. The comprehensiveness of Wikipedia has made the online encyclopedia an increasingly popular target for disambiguation. Disambiguation to Wikipedia is similar to a traditional Word Sense Disambiguation task, but distinct in that the Wikipedia link structure provides additional information about which disambiguations are compatible. In this work we analyze approaches that utilize this information to arrive at coherent sets of disambiguations for a given document (which we call "global" approaches), and compare them to more traditional (local) approaches. We show that previous approaches for global disambiguation can be improved, but even then the local disambiguation provides a baseline which is very hard to beat.

1 Introduction

Wikification is the task of identifying and linking expressions in text to their referent Wikipedia pages. Recently, Wikification has been shown to form a valuable component for numerous natural language processing tasks including text classification (Gabrilovich and Markovitch, 2007b; Chang et al., 2008), measuring semantic similarity between texts (Gabrilovich and Markovitch, 2007a), cross-document co-reference resolution (Finin et al., 2009; Mayfield et al., 2009), and other tasks (Kulkarni et al., 2009).

Previous studies on Wikification differ with respect to the corpora they address and the subset of expressions they attempt to link. For example, some studies focus on linking only named entities, whereas others attempt to link all "interesting" expressions, mimicking the link structure found in Wikipedia. Regardless, all Wikification systems are faced with a key *Disambiguation to Wikipedia* (D2W) task. In the D2W task, we're given a text along with explicitly identified substrings (called *mentions*) to disambiguate, and the goal is to output the corresponding Wikipedia page, if any, for each mention. For example, given the input sentence "I am visiting friends in <Chicago>," we output *http://en.wikipedia.org/wiki/Chicago* – the Wikipedia page for the city of Chicago, Illinois, and not (for example) the page for the 2002 film of the same name.

Local D2W approaches disambiguate each mention in a document separately, utilizing clues such as the textual similarity between the document and each candidate disambiguation's Wikipedia page. Recent work on D2W has tended to focus on more sophisticated *global* approaches to the problem, in which all mentions in a document are disambiguated simultaneously to arrive at a *coherent* set of disambiguations (Cucerzan, 2007; Milne and Witten, 2008b; Han and Zhao, 2009). For example, if a mention of "Michael Jordan" refers to the computer scientist rather than the basketball player, then we would expect a mention of "Monte Carlo" in the same document to refer to the statistical technique rather than the location. Global approaches utilize the Wikipedia link graph to estimate coherence.

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 1375–1384,
Portland, Oregon, June 19-24, 2011.

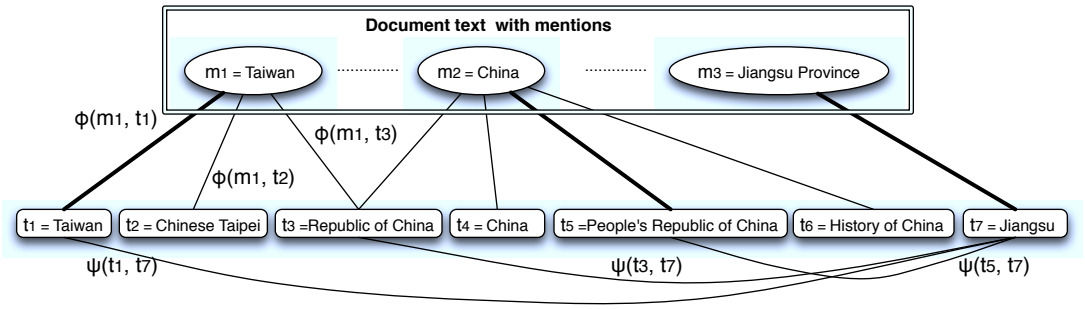

Figure 1: Sample Disambiguation to Wikipedia problem with three mentions. The mention "Jiangsu" is unambiguous. The correct mapping from mentions to titles is marked by heavy edges

In this paper, we analyze global and local approaches to the D2W task. Our contributions are as follows: (1) We present a formulation of the D2W task as an optimization problem with local and global variants, and identify the strengths and the weaknesses of each, (2) Using this formulation, we present a new global D2W system, called GLOW. In experiments on existing and novel D2W data sets,[1] GLOW is shown to outperform the previous state-of-the-art system of (Milne and Witten, 2008b), (3) We present an error analysis and identify the key remaining challenge: determining when mentions refer to concepts *not* captured in Wikipedia.

2 Problem Definition and Approach

We formalize our *Disambiguation to Wikipedia* (D2W) task as follows. We are given a document d with a set of mentions $M = \{m_1, \ldots, m_N\}$, and our goal is to produce a mapping from the set of mentions to the set of Wikipedia titles $W = \{t_1, \ldots, t_{|W|}\}$. Often, mentions correspond to a concept *without* a Wikipedia page; we treat this case by adding a special *null* title to the set W.

The D2W task can be visualized as finding a many-to-one matching on a bipartite graph, with mentions forming one partition and Wikipedia titles the other (see Figure 1). We denote the output matching as an N-tuple $\Gamma = (t_1, \ldots, t_N)$ where t_i is the output disambiguation for mention m_i.

[1]The data sets are available for download at http://cogcomp.cs.illinois.edu/Data

2.1 Local and Global Disambiguation

A *local* D2W approach disambiguates each mention m_i separately. Specifically, let $\phi(m_i, t_j)$ be a score function reflecting the likelihood that the candidate title $t_j \in W$ is the correct disambiguation for $m_i \in M$. A local approach solves the following optimization problem:

$$\Gamma^*_{\text{local}} = \arg\max_{\Gamma} \sum_{i=1}^{N} \phi(m_i, t_i) \quad (1)$$

Local D2W approaches, exemplified by (Bunescu and Pasca, 2006) and (Mihalcea and Csomai, 2007), utilize ϕ functions that assign higher scores to titles with content similar to that of the input document.

We expect, all else being equal, that the correct disambiguations will form a "coherent" set of related concepts. Global approaches define a coherence function ψ, and attempt to solve the following disambiguation problem:

$$\Gamma^* = \arg\max_{\Gamma} [\sum_{i=1}^{N} \phi(m_i, t_i) + \psi(\Gamma)] \quad (2)$$

The global optimization problem in Eq. 2 is NP-hard, and approximations are required (Cucerzan, 2007). The common approach is to utilize the Wikipedia link graph to obtain an estimate pairwise relatedness between titles $\psi(t_i, t_j)$ and to efficiently generate a *disambiguation context* Γ', a rough approximation to the optimal Γ^*. We then solve the easier problem:

$$\Gamma^* \approx \arg\max_{\Gamma} \sum_{i=1}^{N} [\phi(m_i, t_i) + \sum_{t_j \in \Gamma'} \psi(t_i, t_j)] \quad (3)$$

Eq. 3 can be solved by finding each t_i and then mapping m_i independently as in a local approach, but still enforces some degree of coherence among the disambiguations.

3 Related Work

Wikipedia was first explored as an information source for named entity disambiguation and information retrieval by Bunescu and Pasca (2006). There, disambiguation is performed using an SVM kernel that compares the lexical context around the ambiguous named entity to the content of the candidate disambiguation's Wikipedia page. However, since each ambiguous mention required a separate SVM model, the experiment was on a very limited scale. Mihalcea and Csomai applied Word Sense Disambiguation methods to the Disambiguation to Wikipedia task (2007). They experimented with two methods: (a) the lexical overlap between the Wikipedia page of the candidate disambiguations and the context of the ambiguous mention, and (b) training a Naive Bayes classiffier for each ambiguous mention, using the hyperlink information found in Wikipedia as ground truth. Both (Bunescu and Pasca, 2006) and (Mihalcea and Csomai, 2007) fall into the local framework.

Subsequent work on Wikification has stressed that assigned disambiguations for the same document should be related, introducing the global approach (Cucerzan, 2007; Milne and Witten, 2008b; Han and Zhao, 2009; Ferragina and Scaiella, 2010). The two critical components of a global approach are the semantic relatedness function ψ between two titles, and the disambiguation context Γ'. In (Milne and Witten, 2008b), the semantic context is defined to be a set of "unambiguous surface forms" in the text, and the title relatedness ψ is computed as Normalized Google Distance (NGD) (Cilibrasi and Vitanyi, 2007).[2] On the other hand, in (Cucerzan, 2007) the disambiguation context is taken to be all plausible disambiguations of the named entities in the text, and title relatedness is based on the overlap in categories and incoming links. Both approaches have limitations. The first approach relies on the presence of unambiguous mentions in the input document, and the second approach inevitably adds irrelevant titles to the disambiguation context. As we demonstrate in our experiments, by utilizing a more accurate disambiguation context, GLOW is able to achieve better performance.

[2](Milne and Witten, 2008b) also weight each mention in Γ' by its estimated disambiguation utility, which can be modeled by augmenting ψ on per-problem basis.

4 System Architecture

In this section, we present our global D2W system, which solves the optimization problem in Eq. 3. We refer to the system as GLOW, for Global Wikification. We use GLOW as a test bed for evaluating local and global approaches for D2W. GLOW combines a powerful local model ϕ with an novel method for choosing an accurate disambiguation context Γ', which as we show in our experiments allows it to outperform the previous state of the art.

We represent the functions ϕ and ψ as weighted sums of features. Specifically, we set:

$$\phi(m,t) = \sum_i w_i \phi_i(m,t) \tag{4}$$

where each feature $\phi_i(m,t)$ captures some aspect of the relatedness between the mention m and the Wikipedia title t. Feature functions $\psi_i(t,t')$ are defined analogously. We detail the specific feature functions utilized in GLOW in following sections. The coefficients w_i are learned using a Support Vector Machine over bootstrapped training data from Wikipedia, as described in Section 4.5.

At a high level, the GLOW system optimizes the objective function in Eq. 3 in a two-stage process. We first execute a *ranker* to obtain the best non-null disambiguation for each mention in the document, and then execute a *linker* that decides whether the mention should be linked to Wikipedia, or whether instead switching the top-ranked disambiguation to *null* improves the objective function. As our experiments illustrate, the linking task is the more challenging of the two by a significant margin.

Figure 2 provides detailed pseudocode for GLOW. Given a document d and a set of mentions M, we start by augmenting the set of mentions with all phrases in the document that *could* be linked to Wikipedia, but were not included in M. Introducing these additional mentions provides context that may be informative for the global coherence computation (it has no effect on local approaches). In the second

Algorithm: Disambiguate to Wikipedia
Input: document d, Mentions $M = \{m_1, \ldots, m_N\}$
Output: a disambiguation $\Gamma = (t_1, \ldots, t_N)$.
1) Let M' = $M \cup$ { Other potential mentions in d}
2) For each mention $m'_i \in M'$, construct a set of disambiguation candidates $T_i = \{t^i_1, \ldots, t^i_{k_i}\}, t^i_j \neq$ null
3) **Ranker***: Find a solution $\Gamma = (t'_1, \ldots, t'_{|M'|})$, where $t'_i \in T_i$ is the best non-null disambiguation of m'_i.*
4) **Linker***: For each m'_i, map t'_i to* null *in* Γ iff *doing so improves the objective function*
5) Return Γ entries for the original mentions M.

Figure 2: High-level pseudocode for GLOW.

Baseline Feature: $P(t\|m)$, $P(t)$
Local Features: $\phi_i(t, m)$
cosine-sim(Text(t),Text(m)) : Naive/Reweighted
cosine-sim(Text(t),Context(m)): Naive/Reweighted
cosine-sim(Context(t),Text(m)): Naive/Reweighted
cosine-sim(Context(t),Context(m)): Naive/Reweighted
Global Features: $\psi_i(t_i, t_j)$
$I_{[t_i - t_j]}$**PMI(InLinks(t_i),InLinks(t_j)) : avg/max*
$I_{[t_i - t_j]}$**NGD(InLinks(t_i),InLinks(t_j)) : avg/max*
$I_{[t_i - t_j]}$**PMI(OutLinks(t_i),OutLinks(t_j)) : avg/max*
$I_{[t_i - t_j]}$**NGD(OutLinks(t_i),OutLinks(t_j)) : avg/max*
$I_{[t_i \leftrightarrow t_j]}$ *: avg/max*
$I_{[t_i \leftrightarrow t_j]}$**PMI(InLinks($t_i$),InLinks($t_j$)) : avg/max*
$I_{[t_i \leftrightarrow t_j]}$**NGD(InLinks($t_i$),InLinks($t_j$)) : avg/max*
$I_{[t_i \leftrightarrow t_j]}$**PMI(OutLinks($t_i$),OutLinks($t_j$)) : avg/max*
$I_{[t_i \leftrightarrow t_j]}$**NGD(OutLinks($t_i$),OutLinks($t_j$)) : avg/max*

Table 1: Ranker features. $I_{[t_i - t_j]}$ is an indicator variable which is 1 iff t_i links to t_j or vise-versa. $I_{[t_i \leftrightarrow t_j]}$ is 1 iff the titles point to each other.

step, we construct for each mention m_i a limited set of candidate Wikipedia titles T_i that m_i may refer to. Considering only a small subset of Wikipedia titles as potential disambiguations is crucial for tractability (we detail which titles are selected below). In the third step, the ranker outputs the most appropriate non-null disambiguation t_i for each mention m_i.

In the final step, the linker decides whether the top-ranked disambiguation is correct. The disambiguation (m_i, t_i) may be incorrect for several reasons: (1) mention m_i does not have a corresponding Wikipedia page, (2) m_i does have a corresponding Wikipedia page, but it was not included in T_i, or (3) the ranker erroneously chose an incorrect disambiguation over the correct one.

In the below sections, we describe each step of the GLOW algorithm, and the local and global features utilized, in detail. Because we desire a system that can process documents at scale, each step requires trade-offs between accuracy and efficiency.

4.1 Disambiguation Candidates Generation

The first step in GLOW is to extract all mentions that can refer to Wikipedia titles, and to construct a set of disambiguation candidates for each mention. Following previous work, we use Wikipedia hyperlinks to perform these steps. GLOW utilizes an *anchor-title* index, computed by crawling Wikipedia, that maps each distinct hyperlink anchor text to its target Wikipedia titles. For example, the anchor text "Chicago" is used in Wikipedia to refer both to the city in Illinois and to the movie. Anchor texts in the index that appear in document d are used to supplement the mention set M in Step 1 of the GLOW algorithm in Figure 2. Because checking *all* substrings in the input text against the index is computationally inefficient, we instead prune the search space by applying a publicly available shallow parser and named entity recognition system.[3] We consider only the expressions marked as named entities by the NER tagger, the noun-phrase chunks extracted by the shallow parser, and all sub-expressions of up to 5 tokens of the noun-phrase chunks.

To retrieve the disambiguation candidates T_i for a given mention m_i in Step 2 of the algorithm, we query the anchor-title index. T_i is taken to be the set of titles most frequently linked to with anchor text m_i in Wikipedia. For computational efficiency, we utilize only the top 20 most frequent target pages for the anchor text; the accuracy impact of this optimization is analyzed in Section 6.

From the anchor-title index, we compute two local features $\phi_i(m, t)$. The first, $P(t|m)$, is the fraction of times the title t is the target page for an anchor text m. This single feature is a very reliable indicator of the correct disambiguation (Fader et al., 2009), and we use it as a baseline in our experiments. The second, $P(t)$, gives the fraction of all Wikipedia articles that link to t.

4.2 Local Features ϕ

In addition to the two baseline features mentioned in the previous section, we compute a set of text-based

[3] Available at http://cogcomp.cs.illinois.edu/page/software.

local features $\phi(t, m)$. These features capture the intuition that a given Wikipedia title t is more likely to be referred to by mention m appearing in document d if the Wikipedia page for t has high textual similarity to d, or if the context surrounding hyperlinks to t are similar to m's context in d.

For each Wikipedia title t, we construct a top-200 token TF-IDF summary of the Wikipedia page t, which we denote as $Text(t)$ and a top-200 token TF-IDF summary of the context within which t was hyperlinked to in Wikipedia, which we denote as $Context(t)$. We keep the IDF vector for all tokens in Wikipedia, and given an input mention m in a document d, we extract the TF-IDF representation of d, which we denote $Text(d)$, and a TF-IDF representation of a 100-token window around m, which we denote $Context(m)$. This allows us to define four local features described in Table 1.

We additionally compute *weighted* versions of the features described above. Error analysis has shown that in many cases the summaries of the different disambiguation candidates for the same surface form s were very similar. For example, consider the disambiguation candidates of "China' and their TF-IDF summaries in Figure 1. The majority of the terms selected in *all* summaries refer to the general issues related to China, such as *"legalism, reform, military, control, etc."*, while a minority of the terms actually allow disambiguation between the candidates. The problem stems from the fact that the TF-IDF summaries are constructed against the entire Wikipedia, and not against the confusion set of disambiguation candidates of m. Therefore, we *re-weigh* the TF-IDF vectors using the TF-IDF scheme on the disambiguation candidates as a ad-hoc document collection, similarly to an approach in (Joachims, 1997) for classifying documents. In our scenario, the TF of the a token is the original TF-IDF summary score (a real number), and the IDF term is the sum of all the TF-IDF scores for the token within the set of disambiguation candidates for m. This adds 4 more "reweighted local" features in Table 1.

4.3 Global Features ψ

Global approaches require a disambiguation context Γ' and a relatedness measure ψ in Eq. 3. In this section, we describe our method for generating a disambiguation context, and the set of global features $\psi_i(t, t')$ forming our relatedness measure.

In previous work, Cucerzan defined the disambiguation context as the union of disambiguation candidates for all the named entity mentions in the input document (2007). The disadvantage of this approach is that irrelevant titles are inevitably added to the disambiguation context, creating noise. Milne and Witten, on the other hand, use a set of unambiguous mentions (2008b). This approach utilizes only a fraction of the available mentions for context, and relies on the presence of unambiguous mentions with high disambiguation utility. In GLOW, we utilize a simple and efficient alternative approach: we first train a local disambiguation system, and then use the predictions of that system as the disambiguation context. The advantage of this approach is that unlike (Milne and Witten, 2008b) we use all the available mentions in the document, and unlike (Cucerzan, 2007) we reduce the amount of irrelevant titles in the disambiguation context by taking only the top-ranked disambiguation per mention.

Our global features are refinements of previously proposed semantic relatedness measures between Wikipedia titles. We are aware of two previous methods for estimating the relatedness between two Wikipedia concepts: (Strube and Ponzetto, 2006), which uses category overlap, and (Milne and Witten, 2008a), which uses the incoming link structure. Previous work experimented with two relatedness measures: NGD, and Specificity-weighted Cosine Similarity. Consistent with previous work, we found NGD to be the better-performing of the two. Thus we use only NGD along with a well-known Pontwise Mutual Information (PMI) relatedness measure. Given a Wikipedia title collection W, titles t_1 and t_2 with a set of incoming links L_1, and L_2 respectively, PMI and NGD are defined as follows:

$$NGD(L_1, L_2) = \frac{Log(Max(|L_1|, |L_2|)) - Log(|L_1 \cap L_2|)}{Log(|W|) - Log(Min(|L_1|, |L_2|))}$$

$$PMI(L_1, L_2) = \frac{|L_1 \cap L_2|/|W|}{|L_1|/|W||L_2|/|W|}$$

The NGD and the PMI measures can also be computed over the set of *outgoing* links, and we include these as features as well. We also included a feature indicating whether the articles each link to one

another. Lastly, rather than taking the sum of the relatedness scores as suggested by Eq. 3, we use two features: the average and the maximum relatedness to Γ'. We expect the average to be informative for many documents. The intuition for also including the maximum relatedness is that for longer documents that may cover many different subtopics, the maximum may be more informative than the average.

We have experimented with other semantic features, such as category overlap or cosine similarity between the TF-IDF summaries of the titles, but these did not improve performance in our experiments. The complete set of global features used in GLOW is given in Table 1.

4.4 Linker Features

Given the mention m and the top-ranked disambiguation t, the linker attempts to decide whether t is indeed the correct disambiguation of m. The linker includes the same features as the ranker, plus additional features we expect to be particularly relevant to the task. We include the confidence of the ranker in t with respect to second-best disambiguation t', intended to estimate whether the ranker may have made a mistake. We also include several properties of the mention m: the entropy of the distribution $P(t|m)$, the percent of Wikipedia titles in which m appears hyperlinked versus the percent of times m appears as plain text, whether m was detected by NER as a named entity, and a Good-Turing estimate of how likely m is to be out-of-Wikipedia concept based on the counts in $P(t|m)$.

4.5 Linker and Ranker Training

We train the coefficients for the ranker features using a linear Ranking Support Vector Machine, using training data gathered from Wikipedia. Wikipedia links are considered gold-standard links for the training process. The methods for compiling the Wikipedia training corpus are given in Section 5.

We train the linker as a separate linear Support Vector Machine. Training data for the linker is obtained by applying the ranker on the training set. The mentions for which the top-ranked disambiguation did not match the gold disambiguation are treated as negative examples, while the mentions the ranker got correct serve as positive examples.

Mentions/Distinct titles			
data set	Gold	Identified	Solvable
ACE	257/255	213/212	185/184
MSNBC	747/372	530/287	470/273
AQUAINT	727/727	601/601	588/588
Wikipedia	928/813	855/751	843/742

Table 2: Number of mentions and corresponding distinct titles by data set. Listed are (number of mentions)/(number of distinct titles) for each data set, for each of three mention types. *Gold* mentions include all disambiguated mentions in the data set. *Identified* mentions are gold mentions whose correct disambiguations exist in GLOW's author-title index. *Solvable* mentions are identified mentions whose correct disambiguations are among the candidates selected by GLOW (see Table 3).

5 Data sets and Evaluation Methodology

We evaluate GLOW on four data sets, of which two are from previous work. The first data set, from (Milne and Witten, 2008b), is a subset of the *AQUAINT* corpus of newswire text that is annotated to mimic the hyperlink structure in Wikipedia. That is, only the first mentions of "important" titles were hyperlinked. Titles deemed uninteresting and redundant mentions of the same title are not linked. The second data set, from (Cucerzan, 2007), is taken from *MSNBC* news and focuses on disambiguating named entities after running NER and co-reference resolution systems on newsire text. In this case, *all* mentions of all the detected named entities are linked.

We also constructed two additional data sets. The first is a subset of the *ACE* co-reference data set, which has the advantage that mentions and their types are given, and the co-reference is resolved. We asked annotators on Amazon's Mechanical Turk to link the first nominal mention of each co-reference chain to Wikipedia, if possible. Finding the accuracy of a majority vote of these annotations to be approximately 85%, we manually corrected the annotations to obtain ground truth for our experiments. The second data set we constructed, *Wiki*, is a sample of paragraphs from Wikipedia pages. Mentions in this data set correspond to existing hyperlinks in the Wikipedia text. Because Wikipedia editors explicitly link mentions to Wikipedia pages, their anchor text tends to match the title of the linked-to-page—as a result, in the overwhelming majority of

cases, the disambiguation decision is as trivial as string matching. In an attempt to generate more challenging data, we extracted 10,000 random paragraphs for which choosing the top disambiguation according to $P(t|m)$ results in at least a 10% ranker error rate. 40 paragraphs of this data was utilized for testing, while the remainder was used for training.

The data sets are summarized in Table 2. The table shows the number of annotated mentions which were hyperlinked to *non-null* Wikipedia pages, and the number of titles in the documents (without counting repetitions). For example, the AQUAINT data set contains 727 mentions,[4] all of which refer to distinct titles. The MSNBC data set contains 747 mentions mapped to non-null Wikipedia pages, but some mentions within the same document refer to the same titles. There are 372 titles in the data set, when multiple instances of the same title within one document are not counted.

To isolate the performance of the individual components of GLOW, we use multiple distinct metrics for evaluation. *Ranker accuracy*, which measures the performance of the ranker alone, is computed only over those mentions with a non-null gold disambiguation that appears in the candidate set. It is equal to the fraction of these mentions for which the ranker returns the correct disambiguation. Thus, a perfect ranker should achieve a ranker accuracy of 1.0, irrespective of limitations of the candidate generator. *Linker accuracy* is defined as the fraction of *all* mentions for which the linker outputs the correct disambiguation (note that, when the title produced by the ranker is incorrect, this penalizes linker accuracy). Lastly, we evaluate our whole system against other baselines using a previously-employed "bag of titles" (BOT) evaluation (Milne and Witten, 2008b). In BOT, we compare the set of titles output for a document with the gold set of titles for that document (ignoring duplicates), and utilize standard precision, recall, and F1 measures.

In BOT, the set of titles is collected from the mentions hyperlinked in the gold annotation. That is, if the gold annotation is { *(China, People's Republic of China), (Taiwan, Taiwan), (Jiangsu, Jiangsu)*} and the predicted anotation is: { *(China, People's Republic of China), (China, History of China), (Taiwan, null), (Jiangsu, Jiangsu), (republic, Government)*} , then the BOT for the gold annotation is: {*People's Republic of China, Taiwan, Jiangsu*} , and the BOT for the predicted annotation is: {*People's Republic of China, History of China, Jiangsu*} . The title *Government* is not included in the BOT for predicted annotation, because its associate mention *republic* did not appear as a mention in the gold annotation. Both the precision and the recall of the above prediction is 0.66. We note that in the BOT evaluation, following (Milne and Witten, 2008b) we consider all the titles within a document, even if some the titles were due to mentions we failed to identify.[5]

[4]The data set contains votes on how important the mentions are. We believe that the results in (Milne and Witten, 2008b) were reported on mentions which the majority of annotators considered important. In contrast, we used all the mentions.

Generated Candidates k	data sets			
	ACE	MSNBC	AQUAINT	Wiki
1	81.69	72.26	91.01	84.79
3	85.44	86.22	96.83	94.73
5	86.38	87.35	97.17	96.37
20	86.85	88.67	97.83	98.59

Table 3: Percent of "solvable" mentions as a function of the number of generated disambiguation candidates. Listed is the fraction of identified mentions m whose target disambiguation t is among the top k candidates ranked in descending order of $P(t|m)$.

6 Experiments and Results

In this section, we evaluate and analyze GLOW's performance on the D2W task. We begin by evaluating the mention detection component (Step 1 of the algorithm). The second column of Table 2 shows how many of the "non-null" mentions and corresponding titles we could successfully identify (e.g. out of 747 mentions in the MSNBC data set, only 530 appeared in our anchor-title index). Missing entities were primarily due to especially rare surface forms, or sometimes due to idiosyncratic capitalization in the corpus. Improving the number of identified mentions substantially is non-trivial; (Zhou et al., 2010) managed to successfully identify only 59 more entities than we do in the MSNBC data set, using a much more powerful detection method based on search engine query logs.

We generate disambiguation candidates for a

[5]We evaluate the mention identification stage in Section 6.

Features	Data sets ACE	MSNBC	AQUAINT	Wiki
$P(t\|m)$	94.05	81.91	93.19	85.88
$P(t\|m)$+Local				
Naive	95.67	84.04	94.38	92.76
Reweighted	96.21	85.10	**95.57**	**93.59**
All above	95.67	84.68	95.40	93.59
$P(t\|m)$+Global				
NER	96.21	84.04	94.04	89.56
Unambiguous	94.59	84.46	95.40	89.67
Predictions	**96.75**	**88.51**	**95.91**	89.79
$P(t\|m)$+Local+Global				
All features	**97.83**	**87.02**	94.38	**94.18**

Table 4: Ranker Accuracy. Bold values indicate the best performance in each feature group. The global approaches marginally outperform the local approaches on *ranker accuracy* , while combing the approaches leads to further marginal performance improvement.

Data set	Local	Global	Local+Global
ACE	$80.1 \rightarrow 82.8$	$80.6 \rightarrow 80.6$	$81.5 \rightarrow 85.1$
MSNBC	$74.9 \rightarrow 76.0$	$77.9 \rightarrow 77.9$	$76.5 \rightarrow 76.9$
AQUAINT	$93.5 \rightarrow 91.5$	$93.8 \rightarrow 92.1$	$92.3 \rightarrow 91.3$
Wiki	$92.2 \rightarrow 92.0$	$88.5 \rightarrow 87.2$	$92.8 \rightarrow 92.6$

Table 5: Linker performance. The notation $X \rightarrow Y$ means that when linking all mentions, the linking accuracy is X, while when applying the trained linker, the performance is Y. The local approaches are better suited for linking than the global approaches. The linking accuracy is very sensitive to domain changes.

System	ACE	MSNBC	AQUAINT	Wiki
Baseline: $P(t\|m)$	69.52	72.83	82.67	81.77
GLOW Local	75.60	74.39	84.52	90.20
GLOW Global	74.73	74.58	84.37	86.62
GLOW	77.25	74.88	83.94	90.54
M&W	72.76	68.49	83.61	80.32

Table 6: End systems performance - BOT F1. The performance of the full system (GLOW) is similar to that of the local version. GLOW outperforms (Milne and Witten, 2008b) on all data sets.

mention m using an anchor-title index, choosing the 20 titles with maximal $P(t|m)$. Table 3 evaluates the accuracy of this generation policy. We report the percent of mentions for which the correct disambiguation is generated in the top k candidates (called "solvable" mentions). We see that the baseline prediction of choosing the disambiguation t which maximizes $P(t|m)$ is very strong (80% of the correct mentions have maximal $P(t|m)$ in all data sets except MSNBC). The fraction of solvable mentions increases until about five candidates per mention are generated, after which the increase is rather slow. Thus, we believe choosing a limit of 20 candidates per mention offers an attractive trade-off of accuracy and efficiency. The last column of Table 2 reports the number of solvable mentions and the corresponding number of titles with a cutoff of 20 disambiguation candidates, which we use in our experiments.

Next, we evaluate the accuracy of the ranker. Table 4 compares the ranker performance with baseline, local and global features. The reweighted local features outperform the unweighted ("Naive") version, and the global approach outperforms the local approach on all data sets except Wikipedia. As the table shows, our approach of defining the disambiguation context to be the predicted disambiguations of a simpler local model ("Predictions") performs better than using NER entities as in (Cucerzan, 2007), or only the unambiguous entities as in (Milne and Witten, 2008b).[6] Combining the local and the global approaches typically results in minor improvements.

While the global approaches are most effective for ranking, the linking problem has different characteristics as shown in Table 5. We can see that the global features are not helpful in general for predicting whether the top-ranked disambiguation is indeed the correct one.

Further, although the trained linker improves accuracy in some cases, the gains are marginal—and the linker decreases performance on some data sets. One explanation for the decrease is that the linker is trained on Wikipedia, but is being tested on non-Wikipedia text which has different characteristics. However, in separate experiments we found that training a linker on out-of-Wikipedia text only increased test set performance by approximately 3 percentage points. Clearly, while ranking accuracy is high overall, different strategies are needed to achieve consistently high linking performance.

A few examples from the ACE data set help il-

[6] In NER we used only the top prediction, because using all candidates as in (Cucerzan, 2007) proved prohibitively inefficient.

lustrate the tradeoffs between local and global features in GLOW. The global system mistakenly links *"<Dorothy Byrne>, a state coordinator for the Florida Green Party, said ..."* to the British journalist, because the journalist sense has high coherence with other mentions in the newswire text. However, the local approach correctly maps the mention to $null$ because of a lack of local contextual clues. On the other hand, in the sentence *"Instead of Los Angeles International, for example, consider flying into <Burbank> or John Wayne Airport in Orange County, Calif."*, the local ranker links the mention *Burbank* to *Burbank,_California*, while the global system correctly maps the entity to *Bob_Hope_Airport*, because the three airports mentioned in the sentence are highly related to one another.

Lastly, in Table 6 we compare the end system BOT F1 performance. The local approach proves a very competitive baseline which is hard to beat. Combining the global and the local approach leads to marginal improvements. The full GLOW system outperforms the existing state-of-the-art system from (Milne and Witten, 2008b), denoted as M&W, on all data sets. We also compared our system with the recent TAGME Wikification system (Ferragina and Scaiella, 2010). However, TAGME is designed for a different setting than ours: extremely short texts, like Twitter posts. The TAGME RESTful API was unable to process some of our documents at once. We attempted to input test documents one sentence at a time, disambiguating each sentence independently, which resulted in poor performance (0.07 points in F1 lower than the $P(t|m)$ baseline). This happened mainly because the same mentions were linked to different titles in different sentences, leading to low precision.

An important question is why M&W underperforms the baseline on the MSNBC and Wikipedia data sets. In an error analysis, M&W performed poorly on the MSNBC data not due to poor disambiguations, but instead because the data set contains only named entities, which were often delimited incorrectly by M&W. Wikipedia was challenging for a different reason: M&W performs less well on the short (one paragraph) texts in that set, because they contain relatively few of the unambiguous entities the system relies on for disambiguation.

7 Conclusions

We have formalized the *Disambiguation to Wikipedia* (D2W) task as an optimization problem with local and global variants, and analyzed the strengths and weaknesses of each. Our experiments revealed that previous approaches for global disambiguation can be improved, but even then the local disambiguation provides a baseline which is very hard to beat.

As our error analysis illustrates, the primary remaining challenge is determining when a mention does *not* have a corresponding Wikipedia page. Wikipedia's hyperlinks offer a wealth of disambiguated mentions that can be leveraged to train a D2W system. However, when compared with mentions from general text, Wikipedia mentions are disproportionately likely to have corresponding Wikipedia pages. Our initial experiments suggest that accounting for this bias requires more than simply training a D2W system on a moderate number of examples from non-Wikipedia text. Applying distinct semi-supervised and active learning approaches to the task is a primary area of future work.

Acknowledgments

This research supported by the Army Research Laboratory (ARL) under agreement W911NF-09-2-0053 and by the Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0181. The third author was supported by a Microsoft New Faculty Fellowship. Any opinions, findings, conclusions or recommendations are those of the authors and do not necessarily reflect the view of the ARL, DARPA, AFRL, or the US government.

References

R. Bunescu and M. 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), Trento, Italy*, pages 9–16, April.

Ming-Wei Chang, Lev Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: dataless classification. In *Proceedings of the*

23rd national conference on Artificial intelligence - Volume 2, pages 830–835. AAAI Press.

Rudi L. Cilibrasi and Paul M. B. Vitanyi. 2007. The google similarity distance. *IEEE Trans. on Knowl. and Data Eng.*, 19(3):370–383.

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, June. Association for Computational Linguistics.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2009. Scaling wikipedia-based named entity disambiguation to arbitrary web text. In *Proceedings of the WikiAI 09 - IJCAI Workshop: User Contributed Knowledge and Artificial Intelligence: An Evolving Synergy*, Pasadena, CA, USA, July.

Paolo Ferragina and Ugo Scaiella. 2010. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In Jimmy Huang, Nick Koudas, Gareth J. F. Jones, Xindong Wu, Kevyn Collins-Thompson, and Aijun An, editors, *Proceedings of the 19th ACM conference on Information and knowledge management*, pages 1625–1628. ACM.

Tim Finin, Zareen Syed, James Mayfield, Paul McNamee, and Christine Piatko. 2009. Using Wikitology for Cross-Document Entity Coreference Resolution. In *Proceedings of the AAAI Spring Symposium on Learning by Reading and Learning to Read*. AAAI Press, March.

Evgeniy Gabrilovich and Shaul Markovitch. 2007a. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 1606–1611, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Evgeniy Gabrilovich and Shaul Markovitch. 2007b. Harnessing the expertise of 70,000 human editors: Knowledge-based feature generation for text categorization. *J. Mach. Learn. Res.*, 8:2297–2345, December.

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*, CIKM '09, pages 215–224, New York, NY, USA. ACM.

Thorsten Joachims. 1997. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pages 143–151, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. 2009. Collective annotation of wikipedia entities in web text. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 457–466, New York, NY, USA. ACM.

James Mayfield, David Alexander, Bonnie Dorr, Jason Eisner, Tamer Elsayed, Tim Finin, Clay Fink, Marjorie Freedman, Nikesh Garera, James Mayfield, Paul McNamee, Saif Mohammad, Douglas Oard, Christine Piatko, Asad Sayeed, Zareen Syed, and Ralph Weischede. 2009. Cross-Document Coreference Resolution: A Key Technology for Learning by Reading. In *Proceedings of the AAAI 2009 Spring Symposium on Learning by Reading and Learning to Read*. AAAI Press, March.

Rada Mihalcea and Andras Csomai. 2007. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, CIKM '07, pages 233–242, New York, NY, USA. ACM.

David Milne and Ian H. Witten. 2008a. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *In the Wikipedia and AI Workshop of AAAI*.

David Milne and Ian H. Witten. 2008b. Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 509–518, New York, NY, USA. ACM.

Michael Strube and Simone Paolo Ponzetto. 2006. Wikirelate! computing semantic relatedness using wikipedia. In *proceedings of the 21st national conference on Artificial intelligence - Volume 2*, pages 1419–1424. AAAI Press.

Yiping Zhou, Lan Nie, Omid Rouhani-Kalleh, Flavian Vasile, and Scott Gaffney. 2010. Resolving surface forms to wikipedia topics. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1335–1343, Beijing, China, August. Coling 2010 Organizing Committee.