Relational Inference for Wikification

Xiao Cheng **Dan Roth**
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
{cheng88,danr}@illinois.edu

Abstract

Wikification, commonly referred to as Disambiguation to Wikipedia (D2W), is the task of identifying concepts and entities in text and disambiguating them into the most specific corresponding Wikipedia pages. Previous approaches to D2W focused on the use of local and global statistics over the given text, Wikipedia articles and its link structures, to evaluate context compatibility among a list of probable candidates. However, these methods fail (often, embarrassingly), when some level of text understanding is needed to support Wikification. In this paper we introduce a novel approach to Wikification by incorporating, along with statistical methods, richer relational analysis of the text. We provide an extensible, efficient and modular Integer Linear Programming (ILP) formulation of Wikification that incorporates the entity-relation inference problem, and show that the ability to identify relations in text helps both candidate generation and ranking Wikipedia titles considerably. Our results show significant improvements in both Wikification and the TAC Entity Linking task.

1 Introduction

Wikification (D2W), the task of identifying concepts and entities in text and disambiguating them into their corresponding Wikipedia page, is an important step toward supporting deeper textual understanding, by augmenting the ability to ground text in existing knowledge and facilitating knowledge expansion.

D2W has been studied extensively recently (Cucerzan, 2007; Mihalcea and Csomai, 2007; Milne and Witten, 2008; Ferragina and Scaiella, 2010; Ratinov et al., 2011) and has already found broad applications in NLP, Information Extraction, and Knowledge Acquisition from text, from coreference resolution (Ratinov and Roth, 2012) to entity linking and knowledge population (Ellis et al., 2011; Ji et al., 2010; Cucerzan, 2011).

Given a document D containing a set of concept and entity mentions M (referred to later as *surface*), the goal of Wikification is to find the most accurate mapping from mentions to Wikipedia *titles* T; this mapping needs to take into account our understanding of the text as well as background knowledge that is often needed to determine the most appropriate title. We also allow a special NIL title that captures all mentions that are outside Wikipedia.

Earlier approaches treated this task as a word-sense disambiguation (WSD) problem, which was later enhanced with a certain level of global reasoning, but essentially all approaches focused on generic statistical features in order to achieve robust disambiguation. It was shown that by disambiguating to the most likely title for every surface, independently maximizing the conditional probability $\Pr(title|surface)$, we already achieve a very competitive baseline on several Wikification datasets (Ratinov et al., 2011). This strong statistical baseline makes use of the relatively comprehensive coverage of the existing Wikipedia links from surface strings to Wikipedia titles. Although more involved statistical features are required in order to make substantial improvements, global features such as context TF-IDF, better string similarity, etc., statistics-based Wikification systems give a fairly coherent set of disambiguation when sufficient context is available. Consider the following example: *Earth's biosphere*

Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1787–1796,
Seattle, Washington, USA, 18-21 October 2013.

*then significantly altered the atmospheric and other basic physical conditions, which enabled the proliferation of organisms. The **atmosphere** is composed of 78.09% nitrogen, 20.95% oxygen, 0.93% argon, 0.039% carbon dioxide, and small amounts of...*

The baseline system we adopted (Ratinov et al., 2011), one of the best Wikification systems, already disambiguates **atmosphere** correctly to the title *Earth's atmosphere* instead of the more general title *Atmosphere*, making use of the concept *Earth* in its local context to resolve the mention to the more specific title that better coheres with the topic. However, consider the following example:

Ex. 1 "As **Mubarak**, the wife of deposed Egyptian President Hosni Mubarak got older, her influence..."

The bold faced name should be mapped to *Suzanne Mubarak*, but all existing Wikification systems map both names in this sentence to the dominant page (the most linked page) of *Hosni Mubarak*, failing to understand the relation between them, which should prevent them from being mapped to the same page. A certain level of text understanding is required even to be able to generate a good list of title candidates. For example, in:

Ex. 2 "...ousted long time Yugoslav President Slobodan Milošević in October. Mr. Milošević's **Socialist Party**..."

the bold-faced concept should be mapped to the page of the *Socialist Party of Serbia*, which is far down the list of titles that could be related to "Socialist Party"; making this title a likely candidate requires understanding the possessive relation with Milošević and then making the knowledge-informed decision that he is more related to *Socialist Party of Serbia* than any other possible titles. Finally, in

Ex. 3 "James Senn, director of **Robinson College**'s Center for Global Business Leadership at Georgia State University..."

we must link *Robinson College* to *J. Mack Robinson College of Business* which is located at *Georgia State University* instead of *Robinson College, Cambridge*, which is the only probable title linked by the surface *Robinson College* in the version of the Wikipedia dump we used.

These examples further illustrate that, along with understanding the relation expressed in the text, we need to access background knowledge sources and to deal with variability in surface representation across the text, Wikipedia, and knowledge, in order to reliably address the Wikification problem.

In this paper we focus on understanding those natural language constructs that will allow eliminating these "obvious" (to a human reader) mistakes from Wikification. In particular, we focus on resolving coreference and a collection of local syntactico-semantic relations (Chan and Roth, 2011); better understanding the relational structure of the text allows us to generate title candidates more accurately given the text, rank these candidates better and determine when a mention in text has no corresponding title in Wikipedia and should be mapped to NIL, a key problem in Wikification. Moreover, it allows us to access external knowledge based resources more effectively in order to support these decisions.

We incorporate the outcome of our relational analysis, along with the associated features extracted from external sources and the "standard" wikification statistical features, into an ILP-based inference framework that globally determines the best assignment of mentions to titles in a given document. We show that by leveraging a better understanding of the textual relations, we can substantially improve the Wikification performance. Our system significantly outperforms all the top Wikification systems on the widely adopted standard datasets and shows state-of-the-art results when evaluated (without being trained directly) on the TAC 2011 Entity Linking task.

2 The Wikification Approach

A general Wikification decision consists of three computational components: (1) generating a ranked list of title candidates for each mention, (2) ranking candidates globally, and (3) dealing with NIL mentions. For (1), the "standard" way of using $\Pr(title|surface)$ is often not sufficient; consider the case where the mention is the single word "*President*"; disambiguating such mentions depends heavily on the context, i.e. to determine the relevant country or organization. However, it is intractable to search the entire surface-to-title space, and using an arbitrary top-K list will inevitably leave out a large number of potential solutions. For (2), even though

the anchor texts cover many possible ways of paraphrasing the Wikipedia article titles and thus using the top $\Pr(title|surface)$ is proven to be a fairly strong baseline, it is never comprehensive. There is a need to disambiguate titles that were never linked by any anchor text, and to disambiguate mentions that have never been observed as the linked text. For (3) the Wikifier needs to determine when a mention corresponds to no title, and map it to a NIL entity. Simply training a classifier using coherency features or topical models turns out to be insufficient, since it has a predetermined granularity at which it can distinguish entities.

Next we provide a high-level description (Alg. 1) of our approach to improve Wikification by leveraging textual relations in these three stages.

Algorithm 1 Relational Inference for Wikification

*N*ote: $\Gamma : M \rightarrow T$ is the sought after mapping from all mentions in the document to all candidate titles in Wikipedia.

Require: Document D, Knowledge Base K consisting of relation triples $\sigma = (t_a, p, t_b)$, where p is the relation predicate.

1: Generate initial mentions $M = \{m_i\}$ from D.
2: Generate candidates $t_i = \{t_i^k\}$ for mention m_i and initialize candidate priors $\Pr(t_i^k|m_i)$ with existing Wikification system, for all $m_i \in M$.
3: Instantiate non-coreference relational constraints and add relational candidates.
4: Instantiate coreference relational constraints and add relational candidates.
5: Construct an ILP objective function and solve for the $\arg\max_\Gamma \Pr(\Gamma)$.
6: **return** Γ.

Most of our discussion addresses the relational analysis and its impact on stage (2) and (3) above. We will only briefly discuss improvements to the standard candidate generation stage in Sec. 4.4

3 Problem Formulation

We now describe how we formulate our global decision problem as an Integer Linear Program (ILP).

We use two types of boolean variables: e_i^k is used to denote whether we disambiguate m_i to t_i^k ($\Gamma(m_i) = t_i^k$) or not. $r_{ij}^{(k,l)}$ is used to denote if titles t_i^k and t_j^l are chosen simultaneously, that is, $r_{ij}^{(k,l)} = e_i^k \wedge e_j^l$.

Our models determine two types of score for the boolean variables above: $s_i^k = \Pr(e_i^k) = \Pr(\Gamma(m_i) = t_i^k)$, represents the initial score for the kth candidate title being chosen for mention m_i. For a pair of titles (t_i^k, t_j^l), we denote the confidence of finding a relation between them by $w_{ij}^{(k,l)}$. Its value depends on the textual relation type and on how coherent it is with our existing knowledge.

Our goal is to find the best assignment to variables e_i^k, such that it satisfies some legitimacy (hard) constraints and the soft constraints dictated by the relational constraints (via scores $w_{ij}^{(k,l)}$). To accomplish that we define our objective function as a Constrained Conditional Model (CCM) (Roth and Yih, 2004; Chang et al., 2012) that is used to reward or penalize a pair of candidates t_i^k, t_j^l by $w_{ij}^{(k,l)}$ when they are chosen in the same document. Specifically, we choose the assignment Γ_D that optimizes:

$$\Gamma_D = \arg\max_\Gamma \sum_i \sum_k s_i^k e_i^k + \sum_{i,j} \sum_{k,l} w_{ij}^{(k,l)} r_{ij}^{(k,l)}$$

$$\begin{aligned}
s.t. \quad & r_{ij}^{(k,l)} \in \{0,1\} && \text{Integral constraints} \\
& e_i^k \in \{0,1\} && \text{Integral constraints} \\
& \forall i \textstyle\sum_k e_i^k = 1 && \text{Unique solution} \\
& 2r_{ij}^{(k,l)} \leq e_i^k + e_j^l && \text{Relation definition}
\end{aligned}$$

Note that as in most NLP problems, the problem is very sparse, resulting in a tractable ILP that is solved quickly by off-the-shelf ILP packages (Gurobi Optimization, 2013). In our case the key reason for the sparseness is that $w_{ij}^{(k,l)} = 0$ for most pairs considered, which does not require explicit instantiation of $r_{ij}^{(k,l)}$.

4 Relational Analysis

The key challenge in incorporating relational analysis into the Wikification decision is to systematically construct the relational constraints (the solid edges between candidates in Figure 1) and incorporate them into our inference framework. Two main components are needed: first, we need to extract high precision textual relations from the text; then,

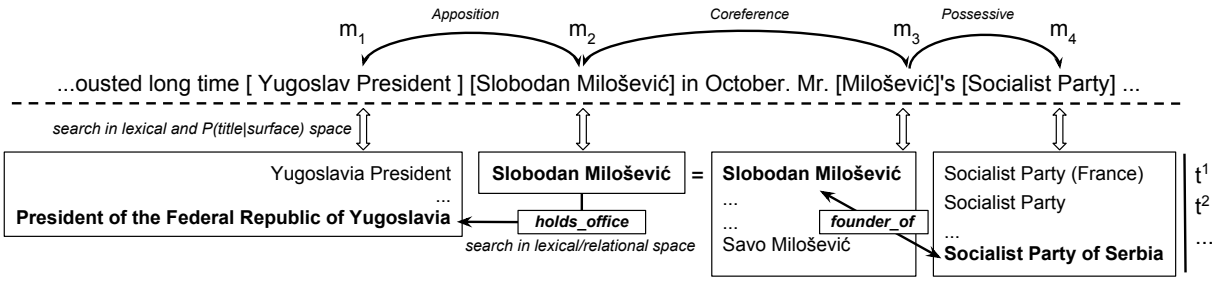

Figure 1: Textual relation inference framework: The goal is to maximize the objective function assigning mentions to titles while enforcing coherency with relations extracted from both text and an external knowledge base. Here, searching the external KB reveals that *Slobodan Milošević* is the founder of the *Socialist Party of Serbia*, which can be referred to by the surface *Socialist Party*; we therefore reward the output containing this pair of candidates. The same idea applies for the relation "*Slobodan Milošević* holds office as *President of the Federal Republic of Yugoslavia*" as well as to the coreference relation between two mentions of *Slobodan Milošević*.

we need to assign weights to these semantic relations. We determine the weights by combining type and confidence of the relation extracted from text with the confidence in relations retrieved from an external Knowledge Base (KB) by using the mention pairs as a query. It is noteworthy that although context window based coherency objective functions capture many proximity relations, using these unfiltered relations as constraints in our experiments introduced excessive amount of false-positives for the intrinsically sparse textual relations and resulted in severe performance hit.

In Sec. 4.1 we describe how we extract relations from text; our goal is to reliably identify *arguments* that we hypothesize to be in a relation; we show that this is essential both to our candidate generation, our ranking and the mapping to NIL. Sec. 4.2 describes how we use an external KB to verify that these arguments are indeed in a relation. Finally, Sec. 4.3 shows how we generate scores for the mentions and relations, as coefficients in the objective function of Sec. 3. The process is illustrated in Figure 1. Overall, our approach is an ambiguity-aware approach that identifies, filters and scores the relevant relations; this is essential due to the ambiguity, variability and noise inherent in directly matching surface forms to titles.

4.1 Relation Extraction

Even though relation extraction is an open problem, analysis on the ACE2004 Relation Detection and Characterization (RDC) dataset shows that approximately 80% of the relations are expressed through syntactico-semantic structures (Chan and Roth, 2011) that are easy to extract with high precision. Unlike the general ACE RDC task, we can restrict relation arguments to be named entities and thus leverage the large number of known relations in existing databases (e.g. Wikipedia infoboxes). We also consider conference relations that potentially aid mapping different mentions to the same title.

4.1.1 Syntactico-semantic Relations

We introduce our approach using the following example. Consider a news article discussing Israeli politics while briefly mentioning:

Ex. 4 An official at the [Iranian]$_1$ [Ministry of Defense]$_2$ told Tehran Radio that...

A purely statistical approach would very likely map the entity *[Ministry of Defense]*$_2$ to *Ministry of Defense (Israel)* instead of *Ministry of Defense and Armed Forces Logistics (Iran)* because the context is more coherent with concepts related to Israel rather than to Iran. Nevertheless, the pre-modifier relation between *[Iranian]*$_1$ and *[Ministry of Defense]*$_2$ demands the answer to be tightly related to Iran. Even though human readers may not know the correct title needed here, understanding the pre-modifier relation allows them to easily filter through a list of candidates and enforce constraints that are derived jointly from the relation expressed in the text and their background knowledge.

In our attempt to mimic this general approach, we employ several high precision classifiers to resolve

a range of local relations that are used to retrieve relevant background knowledge, and consequently integrated into our inference framework. Our input for relation extraction is any segment matched by the regular expression to be mentioned in section 4.4 in the candidate generation stage; we analyze its constituents by decomposing it into the two largest sub-entities that have (in Wikipedia) corresponding candidates. In the above example, *Iranian Ministry of Defense* would be decomposed into **Iranian** and **Ministry of Defense** and our relation extraction process hypothesizes a relation between these arguments.

Note that we do not use any full parsing since it does not address our needs directly nor does it scale well with the typical amount of data used in Wikification.

4.1.2 Coreference Relations

In addition to syntactico-semantic relations, we could also encounter other textual relations. The following example illustrates the importance of understanding co-reference relations in Wikification:

Ex. 5 [Al Goldman]$_1$, chief market strategist at A.G. Edwards, said ... [Goldman]$_2$ told us that...

There is no Wikipedia entry (or redirection) that matches the name *Al Goldman*. Clearly *[Goldman]*$_2$ refers to the same person and should be mapped to the same entity (or to NIL) rather than popular entities frequently referred to as *Goldman*, coherent with context or not, such as *Goldman Sachs*. To accomplish that, we cluster named entities that share tokens or are acronyms of each other when there is no ambiguity (e.g. no other longer named entity mentions containing *Goldman* in the document) and use a voting algorithm (Algorithm 2) to generate candidates locally from within the clusters. We also experimented with using full-fledged coference systems, but found it to be time consuming while providing no significant end-to-end performance difference.

4.1.3 Coreferent Nominal Mentions

Document level coreference also provides important relations between named entities and nominal mentions. Extracting these relations proved to be very useful for classifying NIL entities, as unfamiliar concepts tend to be introduced with these succinct appositional nominal mentions. These descriptions provide a clean "definition" of the entity, allowing us to abstract the inference to a limited "noun phrase entailment problem". That is, it allows us to determine whether the target mention corresponds to a candidate title. Consider, for example, wikifying *Dorothy Byrne* in: **Dorothy Byrne***, a state coordinator for the Florida Green Party, ...*
Identifying the apposition relation allows us to determine that this *Dorothy Byrne* is *not* the baseline Wikipedia title. We use the TF-IDF cosine similarity between the nominal description and the lexical context (Ratinov et al., 2011) of the candidate page, head word attributes and entity relation (i.e. between *Dorothy Byrne* and *Florida Green Party*) to determine whether any candidates of *Dorothy Byrne* can entail the nominal mention.

4.2 Relational Queries

Statistics based candidate generation algorithms always generate the same list of candidates given the same surface string; even though this approach has a competitive coverage rate, it will not work well in some "obvious" (to human) cases; for example, it offers very little information on highly ambiguous surface strings such as "President" for which it is even intractable to rank all the candidates. Top-K lists which were used in previous literature suffer from the same problem. Instead, we make use of relational queries to generate a more likely set of candidates.

Once mention pairs are generated from text using the syntactico-semantic structures and coreference, we use these to query our KB of relational triples. We first indexed all Wikipedia links and DBpedia relations as unordered triples $\sigma = (t_i, p, t_j)$, where the arguments t_i, t_j are tokenized, stemmed and lowercased for best recall. p is either a relation predicate from the DBpedia ontology or the predicate $LINK$ indicating a hyperlink relation. Since our baseline system has approximately 80% accuracy at this stage, it is reasonable to assume that at least one of the argument mentions is correctly disambiguated. Therefore we prune the search space by making only two queries for each mention pair (m_i, m_j): $q_0 = (t_i^*, m_j)$ and $q_1 = (m_i, t_j^*)$ where t_i^*, t_j^* are the strings representing the top titles chosen by the current model for mentions m_i, m_j re-

spectively.

We also aggressively prune the search results in a way similar to the process in Sec. 4.4, only keeping the arguments that are known to be possible or very likely candidates of the mention, based on the ambiguity that exists in the query result.

4.3 Relation Scoring

For the final assignment made using our objective function (Sec. 3) we need to normalize and rescale the output of individual components of our system as they come from different scoring functions. We consider adding new title candidates from two sources, through the coreference module and through the combined DBpedia and Wikipedia inter-page link structures. Next we describe how to compute and combine these scores.

4.3.1 Scoring Knowledge Base Relations

Our model uses both explicit relations $p \neq LINK$ from DBpedia and Wikipedia hyperlinks $p = LINK$ (implicit relation). We want to favor relations with explicit predicate, each weighted as ϕ implicit relation (we use $\phi = 5$ in our experiments, noting the results are insensitive to slight changes of this parameter).

For each query, we denote the score returned by our KB search engine[1] given query q and triple σ as $Sim_{\sigma,q}$. The relational weight $w_{i,j}^{k,l}$ between two candidates (see Sec. 3) is determined as:

$$w_{i,j}^{k,l} = \frac{1}{Z} \sum_{\sigma} \alpha_\sigma Sim_{\sigma,q}$$

where the sum is over the top 20 KB triples, α_σ is the relation type scaling constant (ϕ or 1), and Z is a normalization factor that normalizes all $w_{i,j}^{k,l}$ to the range $[0, 1]$.

Note that we do not check the type of the relation against the textual relation. The key reason is that explicit relations are not as robust, especially considering that we restrict one of the arguments in the relation and constraining the other argument's lexical form. Moreover, we back off to restricting the relations to be between known candidates when multiple lexically matched arguments are retrieved with high ambiguity. Additionally, most of our relations do not have explicit predicates in the text anyhow, and extracting a type would add noise to our decision.

[1] `http://lucene.apache.org/`

Algorithm 2 Coreferent Candidates Voting

Require: Coreference cluster C

Vote collector v_t denotes the score for a candidate t, which by default is 0.
$t_i = \{t_i^1 \dots t_i^n\}$ is the set of candidates of mention m_i.
l_i is the token count of m_i
for all $m_i \in C, l_i \geq 2$ **do**
  for all $t_i^k \in t_i$ **do**
    $v_{t_i^k} = v_{t_i^k} + s_i^k$
  end for
end for
Let $AllSingle$ denote whether $\forall i, l_i = 1$
for all $m_i \in C$ where $l_i = 1$ **do**
  for all $t_i^k \in t_i$ **do**
    if $AllSingle$ or $v_{t_i^k} > 0$ **then**
      $v_{t_i^k} = v_{t_i^k} + s_i^k$
    end if
  end for
end for
return v

4.3.2 Scoring Coreference Relations

For coreference relations, we simply use hard constraints by assigning candidates in the same coreference cluster a high relational weight, which is a cheap approximation to penalizing the output where the coreferent mentions disambiguate to different titles. In practice, using a weight of 10 is sufficient. Another important issue here is that the correct coreferent candidate might not exist in the candidate list of the shorter mentions in the cluster. For example, if a mention has the surface *Richard*, the number of potential candidates is so large that any top K list of titles will not be informative. We therefore ignore candidates generated from short surface strings and give it the same candidate list as the head mentions in its cluster. Figure 2 shows the voting algorithm we use to elect the potential candidates for the cluster.

The reason for separating the votes of longer and shorter mentions is that shorter mentions are inherently more ambiguous. Once a coreferent relation

is determined, longer mentions in the cluster should dictate what this cluster should collectively refer to.

4.4 Candidate Generation

Beyond the algorithmic improvements, the mention and candidate generation stage is aided by a few systematic preprocessing improvement briefly described below.

4.4.1 Mention Segmentation

Since named entities may sometimes overlap with each other, we use regular expressions to match longer surface forms that are often incorrectly segmented or ignored by NER [2] due to different annotation standards. For example, this will capture: *Prime Minister of the United Kingdom*. The regular expression pattern we used for Step 1 in Algorithm 1 simply adds mentions formed by any two consecutive capitalized word chunks connected by up to 2 punctuation marks, prepositions, and the tokens "the", "'s" & "and". These segments are also used as arguments for relation extraction.

4.4.2 Lexical Search

We link certain mentions directly to their exact matching titles in Step 3 when there is very low ambiguity. Specifically, when no title is known for a mention that is relatively long and fuzzily matches the lexically retrieved title, we perform this aggressive linking. The lexical similarity metrics are computed using the publicly available NESim [3] package (Do et al., 2009) with a threshold tuned on a subset of Wikipedia redirects, and by insisting that ORG type entities must have the same head word as the candidate titles. We only accept the link if there exists exactly one title in the lexical searching result after pruning.

5 Experiments and Evaluation

This section describes our experimental evaluation. We compare our system against the top D2W systems and perform several experiments to analyze and better understand the power of our approach. We based our work on the GLOW system from (Ratinov et al., 2011) to initialize the candidates and corresponding priors s_i^k in our objective function. Both the baseline system and our new system are publicly available [4].

5.1 Comparison with other Wikification systems

We first evaluate on the same 4 datasets[5] used in (Ratinov et al., 2011). The AQUAINT dataset, originally introduced in (Milne and Witten, 2008), resembles the Wikipedia annotation structure in that only the first mention of a title is linked, and is thus less sensitive to coreference capabilities. The MSNBC dataset is from (Cucerzan, 2007) and includes many mentions that do not easily map to Wikipedia titles due to rare surface or other idiosyncratic lexicalization (Cucerzan, 2007; Ratinov et al., 2011). Both of these datasets came from the news domain and do not contain any annotated NIL entities. The ACE and Wikipedia datasets are both taken from (Ratinov et al., 2011) where ACE is a subset of ACE2004 Coreference documents annotated by Amazon Mechanical Turkers in a similar standard as in AQUAINT but with NIL entities. The Wikipedia dataset is a sample of Wikipedia pages with its original hyperlink annotation.

The evaluation methodology Bag of Titles (BOT) F1 was used in both (Milne and Witten, 2008; Ratinov et al., 2011). For each document, the gold *bag* of titles is evaluated against our *bag* of system output titles requiring exact segmentation match.

	Dataset			
System	ACE	MSNBC	AQUAINT	Wiki
M&W	72.76	68.49	83.61	80.32
R&R	77.25	74.88	83.94	90.54
RI	**85.30**	**81.20**	**88.88**	**93.09**

Table 1: Performance on Wikification datasets, BOT F1 Performance. Our system, **Relational Inference (RI)** exhibits significant improvements over M&W (Milne and Witten, 2008) and R&R (Ratinov et al., 2011).

[2] We used the IllinoisNER package http://cogcomp.cs.illinois.edu/page/software_view/4

[3] http://cogcomp.cs.illinois.edu/page/software_view/22

[4] http://cogcomp.cs.illinois.edu/page/download_view/Wikifier

[5] http://cogcomp.cs.illinois.edu/page/resource_view/4

5.2 Ablation study

We incrementally add various components to the system and study their impact on the end performance. Due to the changes in Wikipedia since the datasets were generated, some of the pages no longer exist; in order to minimize the interference caused by these inconsistencies to an accurate evaluation of various componenents, we consider all non-NIL gold annotations that do not exist in the current Wikipedia index as NIL entities. Additionally in the MSNBC dataset, 127 out of 756 surface forms are known to be non-recallable. This explains the performance difference between the final rows in Tab. 1 and 2.

	Dataset			
Components	ACE	MSNBC	AQUAINT	Wiki
Baseline	80.68	83.00	83.93	91.93
+Lexical Match	83.47	84.13	88.88	93.41
+Coreference	83.40	87.88	88.88	93.09
RI	85.83	88.16	88.88	93.09

Table 2: Ablation study on Wikification datasets, BOT F1 Performance

The *Baseline* refers to the best performing configuration that was used in (Ratinov et al., 2011) except for using the current Wikipedia redirects. The *Lexical Match* refers to the applying solely the methodology introduced in Sec. 4.4. The *Coreference* performance includes all the inference performed without the KB triples, while the **Relational Inference (RI)** line represents all aspects of the proposed relational inference. It is clear that different datasets show somewhat different characteristics and consequently different gains from the various aspects of our approach but that, overall, all aspects contribute to improved performance.

5.3 TAC Entity Linking 2011

Next we evaluate our approach on the TAC English Entity Linking Task, which provides standardized evaluation metrics, allowing us to compare to a large number of other systems. We did not evaluate on the 2012 English Entity Linking due to the significant amount of ambiguous NIL entities included (Ellis et al., 2011) in the queries and the need to cluster them, which our D2W task definition does not address in depth. We compare our system with the Top 3 TAC 2011 systems (LCC, MS-MLI and NUSchime) as well as our baseline system GLOW that participated in TAC 2011 English Entity Linking (Ratinov and Roth, 2011) in table 3. The evaluation metric is the official modified B^3 and Micro-Average explained in (Ji et al., 2011).

Given the TAC Knowledge Base (TKB), which is a subset of the 2009 Wikipedia Dump, the TAC Entity Linking objective is to answer a named entity query string with either a TKB entry ID or a NIL entity ID, where the NIL entity IDs should be clustered across documents.

It is important to note that we did not retrain our system on the TAC data as the top three systems did, even though the objective function is slightly different. Instead, we ran our system on the TAC documents directly without any query expansion. For the final output of each query, we simply use the most confident candidate among all matched mentions. Due to the clustering requirement, we also trivially cluster NIL entities that either are mapped to the same out-of-KB Wikipedia URL or have the same surface form.

	Performance			
System	MA	B^3 P	B^3 R	B^3 F1
LCC	86.1	84.4	84.7	84.6
MS-MLI	86.8	84.8	83.4	84.1
RI	86.1	82.9	84.5	83.7
NUSchime	86.3	81.5	84.9	83.1
RI-0	81.4	78.6	79.1	78.8
Cogcomp	78.7	75.7	76.5	76.1

Table 3: TAC2011 Entity Linking performance. MA is Micro-Average. LLC (Monahan et al., 2011) is the best performing system in terms of B^3 F1 while MS-MLI (Cucerzan, 2011) is the best in terms of Micro-Average. Cogcomp (Ratinov and Roth, 2011) is the GLOW based system that participated in TAC 2011.**RI** is the complete relational inference system described in this paper; as described in the text, **RI** was not trained on the TAC data, unlike the other top systems.

We performed two runs on the TAC2011 data to study the effects of relational inference. The first run, **RI-0**, uses the current Wikipedia index and

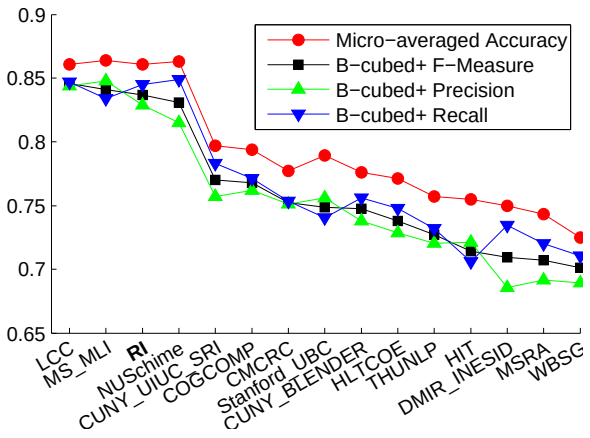

Figure 2: The **RI** compared with the other top 14 TAC2011 English Entity Linking systems ranked by modified B^3 F1 measure. Original figure from (Ji et al., 2011).

redirects for lexical matching without any inference, which scored 2.7% higher than the original GLOW system (Cogcomp). We can regard this performance as the new baseline that benefited from the fuzzy lexical matching capabilities that we have added, as well as the broader set of surface forms and redirects from the current Wikipedia dump. In the second run, **RI**, the complete relational inference described in this paper, scored 4.9% higher than the new baseline and sits on par with the top tier systems despite not being trained on the given data. The LCC system used sophisticated clustering algorithms trained on the TAC development set (Monahan et al., 2011). The second-ranked MS-MLI system relied on topic modeling, external web search engine logs as well as training on the development data (Cucerzan, 2011). This shows the robustness of our methods as well as the general importance of understanding textual relations in the task of Entity Linking and Wikification.

6 Related Work and Discussion

Earlier works on Wikification formulated the task as a WSD problem (Bunescu and Pasca, 2006; Mihalcea and Csomai, 2007) and focused primarily on training a model using local context. Later, various global statistical approaches were proposed to emphasize different coherence measures between the titles of the disambiguated mentions in the same document (Cucerzan, 2007; Milne and Witten, 2008; Ratinov et al., 2011). Built on top of the statistical models, our work focuses on leveraging deeper understanding of the text to more effectively and accurately utilize existing knowledge.

We have demonstrated that, by incorporating textual relations and semantic knowledge as linguistic constraints in an inference framework, it is possible to significantly improve Wikification performance. In particular, we have shown that our system is capable of making "intelligent" inferences that makes use of basic text understanding and has the ability to reason with it and verify it against relevant information sources. This allows our Relational Inference approach to resolve a variety of difficult examples illustrated in the Introduction.

Our system features high modularity since the relations are considered only at inference time; consequently, we can use any underlying Wikification system as long as it outputs a distribution of title candidates for each mention.

One possibility for future work is to supply this framework with a richer set of relations from the text, such as verbal relations. It will also be interesting to incorporate high-level typed relations and relax the relation arguments to be general concepts rather than only named entities.

Acknowledgments

We sincerely thank the three anonymous reviewers for their suggestions on the paper. This material is based on research sponsored by DARPA under agreement number FA8750-13-2-0008, and partly supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20155, by the Army Research Laboratory (ARL) under agreement W911NF-09-2-0053, and by the Multimodal Information Access & Synthesis Center at UIUC, part of CCICADA, a DHS Science and Technology Center of Excellence. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official

policies or endorsements, either expressed or implied, of DARPA, IARPA, DoI/NBC, ARL, or the U.S. Government.

References

R. Bunescu and M. Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the European Chapter of the ACL (EACL)*.

Y. Chan and D. Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Portland, Oregon.

M. Chang, L. Ratinov, and D. Roth. 2012. Structured learning with constrained conditional models. *Machine Learning*, 88(3):399–431, 6.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference of EMNLP-CoNLL*, pages 708–716.

Silviu Cucerzan. 2011. Tac entity linking by performing full-document entity extraction and disambiguation. In *Proceedings of the Text Analysis Conference*.

Q. Do, D. Roth, M. Sammons, Y. Tu, and V. Vydiswaran. 2009. Robust, light-weight approaches to compute lexical similarity. Technical report, Computer Science Department, University of Illinois.

Joe Ellis, Xuansong Li, Kira Griffitt, Stephanie M Strassel, and Jonathan Wright. 2011. Linguistic resources for 2012 knowledge base population evaluations.

Paolo Ferragina and Ugo Scaiella. 2010. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1625–1628, New York, NY, USA. ACM.

Inc. Gurobi Optimization. 2013. Gurobi optimizer reference manual.

Heng Ji, Ralph Grishman, Hoa Trang Dang, Kira Griffitt, and Joe Ellis. 2010. Overview of the tac 2010 knowledge base population track. In *Third Text Analysis Conference (TAC 2010)*.

Heng Ji, Ralph Grishman, and Hoa Trang Dang. 2011. Overview of the tac 2011 knowledge base population track. In *Fourth Text Analysis Conference (TAC 2011)*.

R. Mihalcea and A. Csomai. 2007. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of ACM Conference on Information and Knowledge Management (CIKM)*, pages 233–242.

D. Milne and I. H. Witten. 2008. Learning to link with wikipedia. In *Proceedings of ACM Conference on Information and Knowledge Management (CIKM)*, pages 509–518.

Sean Monahan, John Lehmann, Timothy Nyberg, Jesse Plymale, and Arnold Jung. 2011. Cross-lingual cross-document coreference with entity linking. In *Proceedings of the Text Analysis Conference*.

L. Ratinov and D. Roth. 2011. Glow tac-kbp 2011 entity linking system. In *TAC*. Text Analysis Conference, 11.

L. Ratinov and D. Roth. 2012. Learning-based multi-sieve co-reference resolution with knowledge. In *EMNLP*.

L. Ratinov, D. Roth, D. Downey, and M. Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *ACL*.

D. Roth and W. Yih. 2004. A linear programming formulation for global inference in natural language tasks. In Hwee Tou Ng and Ellen Riloff, editors, *Proceedings of the Annual Conference on Computational Natural Language Learning (CoNLL)*, pages 1–8. Association for Computational Linguistics.