JVN-TDT Entity Linking Systems at TAC-KBP2012

Hien T. Nguyen[a] **Huy H. Minh**[a,b]
[a]Ton Duc Thang University
Nguyen Huu Tho St., Dictrict 7
Ho Chi Minh City, Viet Nam

Tru H. Cao[b,c]
[b]John von Neumann Institute
Linh Trung Ward, Thu Duc District
Ho Chi Minh City, Viet Nam

Trong T. Nguyen[b,c]
[c]Ho Chi Minh City University of Technlogy
268 Ly Thuong Kiet St., Dictrict 10
Ho Chi Minh City, Viet Nam

{hien,huy_hm88}@tdt.edu.vn tru@cse.hcmut.edu.vn nttrong1589@gmail.com

Abstract

We present two methods for entity linking in two of our systems submitted to TAC-KBP 2012. The first one, implemented in JVN-TDT1 system, learns coherence among co-occurrence entities referred to within a text by exploiting Wikipedia's link structure and the second one, implemented in JVN_TDT2 system, combines some heuristics with a statistical model, for entity linking. The method implemented in JVN-TDT1 exploits two features to train a classifier and exploits coreference relations among co-occurring mentions for entity linking. The method implemented in JVN-TDT2 is a hybrid method that performs entity linking in two phases. The first phase is a rule-based phase that filters candidates and, if possible, it disambiguates mentions with high reliability. The second phase employs a statistical model to rank the candidates of each remaining mention and choose the one with the highest ranking as the right referent of that mention. Experiments are conducted to evaluate two methods on two datasets – TAC-KBP2011 and TAC-KBP2012 datasets.

1 Introduction

Entity linking refers to the task of identifying references of entities in a text and linking them to knowledge base entries. It is an essential and challenging component in natural language processing. This paper presents two entity linking methods implemented in our two systems submitted to TAC-KBP 2012. Both of them try to model how human beings disambiguate a mention. When reading a text and encountering a mention, one may rely on his/her knowledge accumulated in the past and the context of the text to identify which one is the underlying entity of a certain mention. Indeed, our methods exploit prior knowledge about entities and analyze the context to perform linking decisions.

Since 2009, entity linking (EL) shared task held at Text Analysis Conference (TAC) (Ji *et al*., 2011; Ji and Grishman, 2011) has attracted more and more attentions in linking entity mentions to knowledge base entries. In EL task, given a query containing a named entity (person, organization, or geo-graphical entity) and a background document including that named entity, an entity linking system is required to provide the ID of the knowledge base (KB) entry to which the name refers; or NIL if there is no such KB entry (Ji and Grishman, 2011). The used KB is Wikipedia.

Wikipedia is a free encyclopedia written by a large number of volunteer contributors. A basic entry in Wikipedia is an *article* that defines and describes a single named entity or a concept. It is uniquely identified by its title that contains a surface form of the corresponding entity and is considered as the ID of that entity. When the surface form is ambiguous, the title may contain further information that we call *title-hint* to distinguish the described entity from others. The title-hint is separated from the surface form by parentheses, e.g.

"John McCarthy (computer scientist)", or a comma, e.g. "Columbia, South Carolina".

In Wikipedia, every article is associated with one or more categories and may have several outgoing links (henceforth *outlinks*) and *redirecting* pages. Each outlink is associated with an anchor text that represents the surface form of the corresponding entity. A redirecting page typically contains only a reference to an article. Title of the redirecting page is an alternative surface form of the described entity or concept in the article. For example, from the redirecting pages of the United States, we extract alternative surface forms of the United States such as "US", "USA", "United States of America", etc.

In this paper, we present our two methods for entity linking. The first one learns coherence among co-occurrence entities referred to within a text by exploiting Wikipedia's link structure and exploits coreference relations among co-occurring mentions to perform entity linking. The second one, published in proceedings of PRICAI 2012 (Nguyen *et al.*, 2012), combines some heuristics with a statistical model for entity linking. These methods are implemented respectively in JVN_TDT1 and JVN_TDT2 systems submitted to TAC-KBP 2012.

2 JVN_TDT1 Entity Linking System

We present the entity linking method implemented in JVN_TDT1 system. In particular, we present two features used this system and how we exploit coreference relations among co-occurring mentions for entity linking. Two used features are prior probability and semantic relatedness. We use prior probability as prior knowledge about entities and use semantic relatedness to estimate relations' strength of co-occurring entities in the same text. A classifier is trained using these two features on a training set consisting of Wikipedia articles. As in Milne and Witten (2008), we train our system on a collection of 500 Wikipedia articles and use 100 other Wikipedia articles that do not appear in the training set to tune the learning parameters.

2.1 Prior probability

Let m be a mention, CE_m be a set of candidate entities of m. Prior probability of an entity $e \in CE_m$ shows the commonness (Medelyan, *et al.*, 2008) of that entity over the others in CE_m. Let $P(e|m)$ denote the prior probability of e given m. The prior probability $P(e|m)$ is defined as follows:

$$P(e \mid m) = \frac{count_m(e)}{\sum_{e_i \in CE_m} count_m(e_i)}$$

where $count_m(e)$ is a function that returns the number of times the mention m is used to refer to entity e in Wikipedia. For instance, assuming that in Wikipedia, a mention m refers to three different entities a, b, and c, in 7, 2, and 1 times respectively; then $P(a|m) = 7/10 = 0.7$, $P(b|m) = 2/10 = 0.2$, $P(c|m) = 1/10 = 0.1$; therefore, a is considered as more popular than b and c given m. Note that $P(e|m)$ of a certain pair of an entity e and a mention m is computed based on Wikipedia as a knowledge base, with different occurrences of m linked to different entities (i.e., Wikipedia articles) including that entity e.

2.2 Semantic relatedness

Co-occurring entities in a text may have relation with each other. Furthermore, the referent of a mention can be inferred from nearby entities that have already been identified. For example, when "Michael Jordan" occurs with "Chicago Bulls" or "NBA", it is more likely that the mention "Michael Jordan" refers to the former player of Chicago Bulls basketball team; meanwhile, if "Georgia" occurs with "Tbilisi" capital as in the text "*TBILISI (CNN) --Most Russian troops have withdrawn from eastern and western Georgia*", it is "Tbilisi" that helps to identify "Georgia" referring to the country next to Russia instead of Georgia state of the US.

Since our target is to estimate how a candidate entity (of a certain mention) relates to co-occurring entities in a text, we measure the strength of relation between that candidate entity and the co-occurring entities in turn. To this end, we adopt the method proposed in (Milne and Witten, 2008) to measure semantic relatedness between two Wikipedia entities based on their ingoing links. In particular, given two entities e_1 and e_2, let A_1 be the set of all Wikipedia articles that link to e_1, A_2 be the set of all Wikipedia articles that link to e_2, and W is the set of all articles in Wikipedia; semantic relatedness between the two entities, e_1 and e_2, called $sem(e_1,e_2)$ is defined as follows:

$$Sem(e_1, e_2) = 1 - \frac{\log(\max(|A_1|, |A_2|)) - \log(|A_1 \cap A_2)|}{\log(|W|) - \log(\min(|A_1|, |A_2|))}$$

2.3 Coreference relation

In reality, an entity may have several different surface forms. Therefore, when referring to a certain entity, one can use one or more of its surface forms. We observed that some surface forms co-occurring in a text and referring to the same entity is common. So this method exploits coreference relations among co-occurring surface forms for entity linking. In particular, we make use of some of orthomatcher rules proposed in (Bontcheva *et al.*, 2002) to identify whether two mentions are coreferent or not.

The way that we exploit coreference relation for entity linking is different from previous work in two folds. Firstly, based on coreference relations, we link an ambiguous mention to an unambiguous mention (or a disambiguated mention). Secondly, we utilize coreference relations to get more candidates in the case when the highest rank among those of candidates of a given mention is not greater than a threshold. For instance, assume that two mentions *m* and *m*' are coreferent and *m* is the mention to be disambiguated; let $\{c_1, c_2, c_3\}$ be the set of candidates of *m* and $\{c'_1, c'_2\}$ be the set of candidates of *m*'; assume that after being ranked, c_1 has the highest rank; if the rank of c_1 is lower than a threshold, our proposed method will rank c'_1 and c'_2 and if the highest rank between those of them is greater than a threshold, *m* is linked to the corresponding candidate. Note that the detected list of candidates for each mention might not be complete; therefore, our method does not require two referent candidates c_i and c_j for *m* and *m'* respectively must be equal.

To produce a reliable coreference relation between two mentions, we prohibit the transitive property. That is because in many cases transitivity in coreference relations causes failure. In particular, assume we know that $\{m_1, m_2\}$ and $\{m_2, m_3\}$ are coreferent pairs, we do not imply that m_1 and m_3 are coreferent. An example in (Bontcheva *et al.*, 2002) showed that if {BBC News, News} and {News, ITV News} are coreferent pairs, {BBC News, ITV News} would be coreferent.

2.4 Linking algorithm

Milne and Witten (2008) employed some classification algorithms to train classifiers using two features mentioned-above. The authors showed that Bagged C4.5 gave the best performance. Similarly, we employ the Bagged C4.5 classification algorithm to train a classifier using the two features: prior probability and semantic relatedness.

```
Algorithm 1 Linking Process using coreference relations
Input: a mention m and its context
Output: a mapping of m and an entity or
     m and NIL
1:  let C be the coreferent set of mention m
2:  let CE be the set of candidates of
     mention m
3:  $e_{top} \leftarrow argmax_{e \in CE}$ BaggedC45(e)
4:  if score[$e_{top}$] ≥ δ then
5:    map m to $e_{top}$
6:  else
7:    let CE′ be the set of candidates
      of all m′ϵ C
8:    $e'_{top} \leftarrow argmax_{e \in CE'}$ BaggedC45(e)
9:    if score[$e'_{top}$] ≥ δ then
10:      map m to $e'_{top}$
11:   else
12:      map m to NIL
13:   end if
14: end if
```

```
Algorithm 2 Linking Process not using coreference relations
Input: a mention m and its context
Output: a mapping of m and an entity or
     m and NIL
1:  let CE be the set of candidates of
     mention m
2:  $e_{top} \leftarrow argmax_{e \in CE}$ BaggedC45(e)
3:  if score[$e_{top}$] ≥ δ then
4:    map m to $e_{top}$
5:  else
6:    map m to NIL
7:  end if
```

FIGURE 1: Linking Algorithms

Figure 1 presents linking algorithms; each of which takes a mention and its context as input. As the same as Milne and Witten (2008) did, we consider mentions that having only one candidate as unambiguous mentions. Given a mention *m* and the text where it occurs, the context of that mention is the set of unambiguous mentions occurring in that text. In linking algorithms presented in Figure 1, the mention *m* is represented by two features presented above.

In Figure 1, Algorithm 1 presents our proposed linking method using coreference relations. Given a mention *m*, let *CE* be the set of candidates of *m*. The classifier Bagged C4.5 is used to rank entities in *CE* (Line 3). If the value of the highest rank entity, say *score*, returned by Bagged C4.5 is greater than a threshold, *m* is mapped to that entity (Line 4-6); otherwise, candidates of mentions that are coreferent with *m* are ranked. If the score of the entity having the highest rank is greater than the threshold, *m* is mapped to that entity; otherwise, *m* is mapped to NIL (Line 9-14). Algorithm 2 presents the linking method that does not use coreference relations. In other words, Algorithm 2 implements the method proposed by Milne and Witten (2008) using two features that are prior probability and semantic relatedness and not using coreference relations among co-occurrence mentions.

2.5 Evaluation

The evaluation metrics we use are micro-average accuracy (MAA) and B-Cubed+ (Ji *et al.*, 2011). We firstly evaluate JVN_TDT1 on TAC-KBP2011 dataset. This dataset consists of 2,250 entity mention queries, in which 1,124 entity mentions refer to entities described by Wikipedia articles. Note that in the following tables, *P* stands for prior probability feature and SR stands for semantic relatedness feature.

Feature	All	NIL	Non-NIL
P	75.3%	87.8%	62.8%
P+SR	**82.5%**	**95.0%**	69.9%

TABLE 1 - The MAA overall results TAC-KBP2011 dataset using coreference

Feature	All	NIL	Non-NIL
P	72.7%	85.0%	61.5%
P+SR	**79.5%**	**91.3%**	68.4%

TABLE 2 - The B-Cubed+ F1 overall results TAC-KBP2011 dataset using coreference

Table 1 and Table 2 present MAA and B-Cubed+ overall results on TAC-KBP2011 dataset using Algorithm 1. Table 3 and Table 4 present MAA and B-Cubed+ overall results on TAC-KBP2011 dataset using Algorithm 2. Table 3 and Table 4 show that if we do not use coreference relations among co-occurring mentions, the performance significantly decreases. In other words, using coreference relations among co-occurring mentions improves about 10% in the best cases when combining prior probability and semantic relatedness for training the classifier. The results in Table 1, 2, 3 and 4 also show that using coreference relations among co-occurring mentions to get more candidate entities improve the performance mainly on non-NIL cases.

Feature	All	NIL	Non-NIL
P	68.3%	90.6%	46.0%
P+SR	72.7%	**96.6%**	48.7%

TABLE 3 - The MAA overall results TAC-KBP2011 dataset not using coreference

Feature	All	NIL	Non-NIL
P	65.5%	87.6%	44.9%
P+SR	69.6%	93.0%	47.3%

TABLE 4 - The B-Cubed+ F1 overall results TAC-KBP2011 dataset not using coreference

We then evaluate our proposed method on TAC-KBP2012 dataset. This dataset consists of 2,226 entity mention queries, in which 1,117 entity mentions refer to entities described by Wikipedia articles. Because Algorithm 1 outperforms Algorithm 2 on TAC-KBP2011 dataset, we run only Algorithm 1 on TAC-KBP2012 dataset. Table 5 presents the highest and median B-Cubed+ F1 of all 94 systems submitted to TAC-KBP 2012.

Query	Highest	Median
All (2,226)	73.0%	53.6%
NIL (1,049)	84.7%%	59.4%
Non-NIL (1,177)	68.7%	49.6%

TABLE 5 - The highest and median B-Cubed+ F1 of 94 systems submitted to TAC-KBP 2012

Query	MAA	B-Cubed+ F1
All (2,226)	67.7%	58.6%
NIL (1,049)	84.9%	71.0%
Non-NIL (1,177)	52.4%	49.8%

TABLE 6 - The MAA and B-Cubed+ F1 overall results TAC-KBP2012 dataset using coreference

Table 6 presents MAA and B-Cubed+ overall results on TAC-KBP 2012 dataset using Algorithm 1. JVN_TDT1 is ranked 7th among all 94 English entity-linking systems submitted to TAC-KBP 2012, and 5th among 40 English entity-linking systems submitted systems to TAC-KBP 2012 and did not use the wiki text element of the reference KB.

3 JVN_TDT2 Entity Linking System

In this Section, we present our entity linking method implemented in JVN_TDT2 system. It is incremental and contains two phases. The first phase is a rule-based phase that filters candidates and, if possible, it disambiguates mentions with high reliability. The second phase employs a statistical model to rank the candidates of each remaining mention and choose the one with the highest ranking as the right referent of that mention. The incremental mechanism of our method is similar to the way humans do when disambiguating mentions based on previously known ones. That is, the proposed method exploits both the flow of information as it progresses in a text, particular in news articles, and the way humans read and understand which entities that the mentions refer to. Indeed, an named entity occurring first in a news article is usually introduced in an unambiguous way, except when it occurs in the headline of the news article. Like humans, our method disambiguates named entities in turn from the top to the bottom of the text. When the referent of a mention is identified, it is considered as an anchor and its identifier and own features are used to disambiguate others. Also, when encountering an ambiguous mention, a reader usually links it to the previously resolved entities and his/her background knowledge to identify what entity that mention refers to. Similarly, our method exploits the coreference chain of mentions in a text and a knowledge base for resolving ambiguous mentions. Furthermore, both humans and our method explore contexts in several levels, from a local one to the whole text, where diverse clues are used for the disambiguation task.

Firstly, we present heuristics employed in the first phase. Secondly, we present a statistical ranking model that is employed in the second phase to rank candidate entities of a mention. Then we present the incremental algorithms in two phases of entity linking. Finally, we present experiments.

3.1 Heuristics

We present main heuristics, namely H_1 and H_2, used in the first phase and based on local contexts of mentions to identify their correct referents. The local context of a location mention is its preceding and succeeding mentions in the text. For example, if "Paris" is a location mention and followed by "France", then the country France is in the local context of this "Paris". The local context of a person or an organization mention comprises the keywords and unambiguous mentions occurring in the same sentence where the mention occurs. We exploit such a local context of a mention to narrow down its candidates and disambiguate its referents if possible.

Let m be the mention to be disambiguated. These two heuristics are stated as follows:

– H_1: Among candidate entities of m, the ones whose title-hints occur around m in a context window are chosen. For instance, given the sentence "A state of emergency has been declared in the US state of Georgia after two people died in storms, a day after a tornado hit the city of Atlanta." for the mention "Atlanta", the candidate entity having the title "Atlanta, Georgia" is chosen because its title-hint "Georgia" occurs around the mention; and for the mention "Georgia" the candidate entity having the title "Georgia (U.S state)" is chosen because its title-hint "US state" occurs around it.

– H_2: if m is a title-hint of an already disambiguated entity around it, the chosen candidate entities are the ones that have outlinks to the disambiguated entity or this disambiguated entity has outlinks to these candidates. For instance, given the phrase "Atlanta, Georgia", after applying H_1, the mention "Atlanta" is annotated with the title "Atlanta, Georgia" in Wikipedia. For the mention "Georgia", after applying H_2, it is annotated with the title "Georgia (U.S state)" in Wikipedia because both Wikipedia articles "Atlanta, Georgia" and "Georgia (U.S state)" have reciprocal links to each other.

3.2 A statistical ranking model

We present a statistical ranking model in the second phase where we employ the Vector Space Model (VSM) to represent mentions in a text and entities in Wikipedia by their features.

The features are divided into two groups: text features extracted from the text and Wikipedia features extracted from Wikipedia articles. The Wikipedia features representing an entity contain the entity title, titles of its redirecting pages, its category labels, its outlink labels.

The text features contain the following types:

– Entity mentions: Each mention identified in the text is considered as a feature. If a mention occurs many times in the text, we keep only one and remove the others. For instance, if "U.S" occurs twice in a text, we remove one.

– Local words: All the words, not including special tokens such as $, #, ?, etc., found inside a specified context window around the mention to be disambiguated. Those local words are not part of mentions occurring in the window context to avoid duplicate features.

- Coreferential words: All local words of the mentions that are co-referent with the mention to be disambiguated in the text. For instance, if “John McCarthy” and “McCarthy” co-occur in the same text and are co-referent, we extract not only words around “John McCarthy” but also those around “McCarthy”.
- IDs: All identifiers of the entities whose mentions have already been linked.

After extracting features for a mention in a text or an entity described in Wikipedia, we put them into a bag-of-words. Then we normalize the bag of words as follows. (i) Removing special characters in some tokens such as normalizing U.S to US, D.C (in “Washington, D.C” for instance) to DC, and so on; (ii) removing punctuation marks and special tokens such as commas, periods, question mark, $, @, etc.; and (iii) removing stop words such as *a*, *an*, *the*, etc., and stemming words using Porter stemming algorithm. The VSM considers the set of features of entities as a bag-of-words. TF-IDF is used to weigh terms and cosine is used to calculate the similarity between feature vectors of mentions and entities.

3.3 Algorithm

In Figure 2, Algorithm 3 takes as an input a set of mentions and returns a set *E* containing mentions that are disambiguated by the proposed heuristics. During the disambiguation process, if a mention is disambiguated, the entity corresponding with it is immediately used to disambiguate the others. The function *revised* (.) makes use of coreference relations among mentions to adjust the linking results. For example, assume that in a text there are occurrences of coreferent mentions “Denny Hillis” and “Hillis”, where “Hillis” may refer to different people such as American actress `Ali Hillis` or American inventor `W. Daniel Hillis`; if “Denny Hillis” is recognized as referring to `W. Daniel Hillis` in Wikipedia, then “Hillis” also refers to `W. Daniel Hillis.`

Note that, to propagate the linking result of a mention to others in its coreference chain, our method checks whether that mention satisfies one of the following criteria: (i) the mention occurs in the text prior to all the others and one of the longest mentions in its coreference chain, or (ii) The mention occurs in the text prior to all the others in its coreference chain and is the main alias of the corresponding referent in Wikipedia. A mention is considered as the main alias of a referent if it occurs in the title of the entity page that describes the corresponding entity in Wikipedia. For example, “United States” is the main alias of the referent the `United States` because it is the title of the entity page describing the United States.

```
Algorithm 3 Heuristics-Based Disambiguation
let $\mathcal{N}$ be a set of mentions
$E \leftarrow \emptyset$
flag ← false
loop until $\mathcal{N}$ empty or flag is true
    $\mathcal{N}' \leftarrow \mathcal{N}$
    for each $n \in \mathcal{N}'$ do
        $C \leftarrow$ a set of candidate entities of $n$
        apply $H_1$, $H_2$ respectively for $n$
        if sizeof(C) = 1 then
            map $n$ to $\gamma^*$  //annotated n with $\gamma^*$
            $E \leftarrow revised(E \cup \{<n \rightarrow \gamma^*>\})$
            remove $n$ from $\mathcal{N}$
        end if
    end for
    if E no change then flag = true
end loop
```

```
Algorithm 4 Statistics-Based Disambiguation
let $\mathcal{N}$ be a set of mentions
$E \leftarrow \emptyset$
flag ← false
loop until $\mathcal{N}$ empty or flag is true
    $\mathcal{N}' \leftarrow \mathcal{N}$
    for each $n \in \mathcal{N}'$ do
        $C \leftarrow$ a set of candidate entities of $n$
        for each candidate c do
            score[c]← Sim(FV(c), FV(m))
        end for
        $\gamma^* \leftarrow \arg\max_{c_i \in C} score[c_i]$
        if $score[\gamma^*] > \tau$ then
            map $n$ to $\gamma^*$   //annotated n with $\gamma^*$
            $E \leftarrow revised(E \cup \{<n \rightarrow \gamma^*>\})$
            remove $n$ from $\mathcal{N}$
        end if
    end for
    if E no change then flag = true
end loop
```

FIGURE 2: Algorithms in two-stage linking process.

In Figure 2, Algorithm 4 takes as an input a set of mentions and returns a set *E* containing disambiguated mentions. The function *FV*(.) at Line 9 of the algorithm employs the VSM where a mention and its candidate entities are represented as bag-of-words as described above. The function *sim*(.) calculates cosine similarity between two feature vectors each of which corresponds to a bag-of-words.

3.4 Evaluation

For a query that contains a mention and a document where the mention occurs, we perform some pre-processing steps on the document. In particular, we perform NE recognition and NE coreference resolution using natural language processing resources of an Information Extraction engine based on GATE (Cunningham *et al.*, 2002), a general architecture for developing natural language processing applications. After these pre-processing steps, we run Algorithm 3 and Algorithm 4 respectively to perform entity linking for all mentions identified in the document.

The evaluation metrics we use are micro-average accuracy (MAA) and B-Cubed+ (Ji *et al.*, 2011). We evaluate JVN_TDT2 on TAC-KBP2011 dataset and TAC-KBP2012 dataset. The results are showed in Table 7 and Table 8 respectively.

Query	MAA	B-Cubed+ F1
All (2,250)	75.8%	72.8%
NIL (1,126)	93.7%	90.4%
Non-NIL (1,124)	57.8%	56%

TABLE 7 - The MAA and B-Cubed+ F1 overall results of JVN_TDT2 on TAC-KBP2011 dataset

Query	MAA	B-Cubed+ F1
All (2,226)	57.1%	47%
NIL (1,049)	75.7%	60.9%
Non-NIL (1,177)	40.5%	37.5%

TABLE 8 - The MAA and B-Cubed+ F1 overall results of JVN_TDT2 on TAC-KBP2012 dataset

4 Conclusions

Entity linking is an essential task in natural language processing applications such as semantic web, information retrieval, question answering, or knowledge base population. This paper presents two methods that link entity mentions in a text to entries of a given knowledge base. The first method learns coherence among co-occurrence entities referred to within a text by exploiting Wikipedia's link structure for entity linking. The second method combines heuristics with a statistical model in an incremental linking process. Experiments are conducted to evaluate two methods on two datasets – TAC-KBP2011 and TAC-KBP2012 datasets. The experiment results show that just exploiting two features – prior probability and semantic relatedness – JVN_TDT1 entity linking system can achieve good performance and coreference relations among mentions significantly contribute to the performance of entity linking systems. The experiments also show that the proposed heuristics are potential for improving the performance of entity linking systems.

References

Ji, H., Grishman, R., and Dang, H. T. (2011). An Overview of the TAC2011 Knowledge Base Population Track. In *Proc. of Text Analysis Conference*.

Ji, H. and Grishman, R. (2011). Knowledge Base Population Successful Approaches and Challenge, in *Proc. of the 49th ACL*, pp. 1148-1158.

Milne, D. and Witten, I.H. (2008). Learning to Link with Wikipedia. In: *Proc. of the 17th ACM CIKM* (CIKM 2008), pp. 509-518.

Medelyan, O., Witten, I. H., Milne, D. (2008). Topic indexing with Wikipedia. In *Proc. of Wikipedia and AI workshop at the AAAI-2008 Conference*.

Bontcheva, K., Dimitrov, M., Maynard, D., Tablan, V., and Cunningham, H. (2002). Shallow Methods for Named Entity Coreference Resolution. In *Proc. of TALN 2002 Workhop*.

Hien T. Nguyen, Tru H. Cao, Trong T. Nguyen, Thuy-Linh Vo-Thi. (2012). Heuristics- and Statistics-based Wikification. In *Proc. of PRICAI 2012*, pp. 879-882.

H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, in *Proc. of the 40th ACL*, 2002.