International Journal of Computational Linguistics and Applications, vol. 5, no. 2, 2014, pp. 49–66
Received 2/11/2014, accepted 10/12/2014, final 15/12/2014.
ISSN 0976-0962, http://ijcla.bahripublications.com

Exploring Linguistic Features for Named Entity Disambiguation

SHUANGSHUANG ZHOU, CANASAI KRUENGKRAI,
NAOAKI OKAZAKI, AND KENTARO INUI

Graduate School of Information Sciences, Japan

ABSTRACT

Resolving named entities is important for a number of natural language processing applications. However, a named entity has multiple name variations while different entities could share the same surface. State-of-the-art systems are based on a global resolution method and mostly adopt link-based features that leverage relationships of co-occurring entities in the knowledge. We found that linguistic features can also significantly affect disambiguation. In this work, we try to explore important linguistic features from context, which could be the fundamental part of the combination of global resolution method and effective features. Therefore, we study and compare the effects of linguistic features in a comprehensive way. Moreover, we found effective linguistic features according to the experiment results.

KEYWORDS: *Named entity disambiguation, entity linking, feature study.*

1 INTRODUCTION

In natural language processing, named entities are important components. However, due to various ways of writing, named entities have multiple surfaces in texts, e.g., Big Blue and IBM. Moreover, different entities often share the same surface. For example, "New York" as a place name has dozens of different referents in Wikipedia.[1] Thus, resolving name mentions to their corresponding entities is necessary. Named entity disambiguation is the task of identifying whether a mention refers to a certain entity and linking mentions to their corresponding entries in a large-scale knowledge base. Therefore, NED (Named entity disambiguation) task is also known as entity linking task.

Ji et al. [1] summarized two main processes: candidate generation and candidate ranking. Because of the variety of named entities, when given a mention, NED systems firstly often generate a candidate list contains as much as possible candidate entities. Then selecting a correct entity from the ranked candidate list by using a ranking model is the final purpose. Since the selected entity should be coherent with the context in the source document, disambiguating entities by leveraging the context information is a fundamental way [2]. Erbs et al. [3] mentioned that features for candidate ranking could be grouped into: linguistic-based (text in source document and text extracted from the KB titles) and link-based (how entities in the same context link in the knowledge).

State-of-the-art systems [4, 2] simultaneously resolve multiple entities (global inference) and mostly adopt link-based features. Those link-based features measure the relatedness of co-occurring entities in context. They assume that a candidate entity could be linked if it and its neighbor entities strongly connect in the knowledge base. For example, in the text of Figure 1, entities in context like *San Antonio Spurs* and *National Basketball Association* strongly support the [Alvin Robertson] candidate for the mention *Robertson* because they are linked in the KB. In Wikipedia, articles titled *San Antonio Spurs* and *National Basketball Association* have in-links from the article titled *Alvin Robertson*. At the same time, the article titled *Alvin Robertson* also have out-links to articles titled *San Antonio Spurs* and *National Basketball Association*.

However, when there is seldom co-occurring entities in context, linguistic information could affect disambiguation significantly [5, 6]. For example, in the text of Figure 2, words like *sophomore*, *British universities*, and *U.S. schools* suggest [University of St.Andrew] as the correct entity for *St.Andrew*. Therefore, we could capture linguistic features, such as comparing the topic distribution of source documents and the KB texts.

We find that linguistic features can locally measure the coherence between mentions and entities in context. Therefore, we study the effects of multiple linguistic features in a comprehensive way in this paper. Espe-

[1] http://en.wikipedia.org/

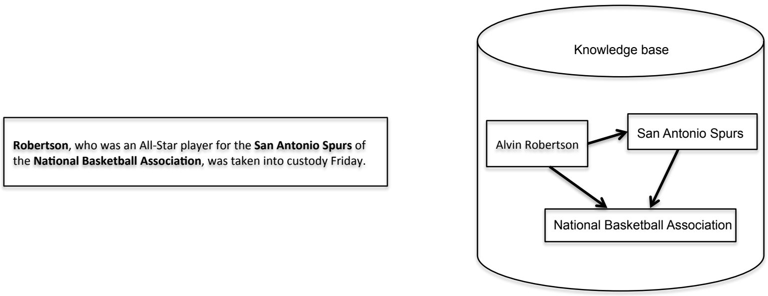

Fig. 1. Example of documents containing mentions for link-based features

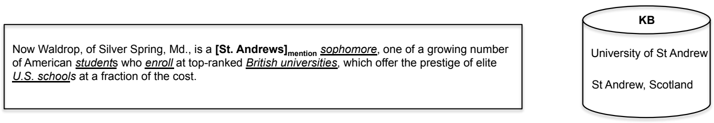

Fig. 2. Example of documents containing mentions for linguistic features

cially, we compare the effectiveness of each linguistic feature, e.g., document similarity, document topics, entity-level features, and POS features.

Moreover, several linguistic features are used as local methods by state-of-the-art global inference systems [4, 2, 7–10]. Considering their insufficient local methods, in this paper, we examine the contributions of linguistic features in order to explore more effective local methods for global inference methods.

2 RELATED WORK

Linguistic features showed promising results in previous studies [5, 6, 11, 12], such as document similarity, word overlapping, entity-level word overlapping, document topics. However, only partial linguistic features are explored by previous work. Dredze et al. [6] captured features based on mentions, source documents and KB entries, but features about document topics are not involved. Zhang et al. [11] made big efforts on candidate selection and acronym expansion, but their disambiguation method only depended on document topics. Therefore, we summarize and refine effective linguistic features of previous work, and propose a broad range of linguistic features in this paper.

Although some previous work reviewed various ranking methods (unsupervised or supervised) and evaluation results [13, 1], they lack comparing effects of linguistic features systemically. Moreover, Garcia et al. [14] systemically reviewed and evaluated several state-of-the-art link-based approaches, but they did not mention linguistic-based context features. To our knowledge, previous work did not examine linguistic features comparatively. Therefore, we try to explore context information on the linguistic level in a comprehensive way.

On the other hand, link-based features strongly depend on the link structure of knowledge base (Wikipedia), e.g., link statistics (incoming links and outgoing links), and category information, etc. Link-based features are mostly used by global inference systems for candidate ranking. *Relatedness* is widely used by [8, 15–17, 2, 4], which is to compute the similarity between two KB entries based on the in/out links. *Relatedness* is effective to measure the relationship between candidates and co-occurring entities in context.

Linguistic features are used to measure the coherence between mentions and candidates, which are also called local methods by previous studies [2, 7, 18, 9, 10]. Combining local methods with global features or global ranking methods, the NED system performance is improved significantly [2]. Among of them, TF/IDF cosine similarity is mostly used by global inference systems for multiple purposes: ranking candidates [2, 7], filtering out noisy candidate [18], and assigning an initial confidence score for subsequent ranking phrase [9, 10]. However, TF/IDF cosine similarity is insufficient to capture the coherence between mentions and entities. Moreover, entity popularity is a salient measure of mentions and entities [8, 19–21], and it could check how likely a surface refer to an entity. Entity popularity is a strong baseline for entity linking [14]. However, this feature could ignore unpopular correct entities.

Therefore, our further motivation is to explore useful linguistic features for global inference. Based on our local features, graph-based methods are applied on short and high-coverage candidate lists, at the same time, unpopular entities can not be missed the candidate list.

3 System Architecture

The TAC KBP entity linking task provides high-quality data set and comprehensive evaluation. The data set contains a query file, a collection of source documents, and a reference KB.

In mention query files, information about one mention is given: the name surface, the document ID, and the position of this mention in the source document (UTF-8 character offsets). For example,

```
<query id="EDL14_ENG_TRAINING_3091">
    <name>St. Andrews</name>
    <docid>WPB_ENG_20101221.0031</docid>
    <beg>1123</beg>
    <end>1133</end>
</query>
```

The example texts in Figure 2 are from source documents. The TAC KBP official reference KB is extracted from an October 2008 dumps of English Wikipedia and consists of 818,741 entries.

Systems are required to generate a link-ID file, which contains pairs of a query and the resolved result (corresponding KB entry ID or NIL). For example, system should output a KB ID e.g., "E0127848" or NIL for the query "EDL14_ENG_TRAINING_3091". In this task, NIL means mentions that do not have entries in the KB. TAC KBP added the mention detection task in 2014. A system should detect possible mentions in raw documents.

We built a pipeline system for this task.[2] The system consists of basic components: mention detection, candidate generation, candidate ranking, NIL classification and NIL clustering. These five components are commonly required for performing Entity Discovery and Linking (EDL) [22]. We add the candidate pruning process after candidate generation to eliminate noisy candidates. In this work, since we want to eliminate the effect of the performance of mention detection, we train and test on the gold mention data set and start from the candidate generation phase. In order to simplify the evaluation, we remove the NIL clustering process.

3.1 *Candidate Generation*

In the candidate generation phase, we need a high-recall candidate list for each mention. In this phase, recall means the percentage of mentions that have the correct entity in the candidate list. If the correct entity does not exist in the candidate list, the ranking process will be in vain. We first group mentions in the source document to handle misspelling, abbreviation, and partial names. For example, the candidate mentions *Gretzy* and

[2] We submitted the system to TAC KBP 2014 entity discovery and linking shared task.

Wayne Gretzky occur in the same source document, and they likely refer to the same entity. If we search candidates by using both of them, the possibility of correct entity appearing in the candidate list of *Gretzy* could be increased.

Moreover, we construct a name variation database, SurfaceSet, by extracting entity title-surface pairs from various Wikipedia sources, such as disambiguation pages, redirection pages, and anchor texts. For example, we extracted name variations like *Barcodes*, *Toon*, *mags*, *magpies*, and *Newcastle* for *Newcastle United F.C.*, a famous England football club. SurfaceSet contains 548,084 entities and 2,080,491 surfaces.

We process one mention at one time. For each mention, we search both the original mention and the same group mentions. We achieved 98.43% recall on the training set. The average number of candidate per list is 245.

3.2 *Candidate Pruning*

Note that the initial candidate lists are too noisy because we want to find as many as possible candidates in the previous phase. Ranking document similarities between source documents and wiki texts is a simple and efficient way to eliminate noisy candidates. According to our preliminary experiment, we found that Latent Semantic Index (LSI) is superior to TF/IDF cosine similarity. LSI achieved 97.39% recall on the training set while TF/IDF got 74.84%. We apply Latent Semantic Index (LSI) to rank each candidate list and retain the top 50 candidates as the final candidate list. We use an off-the-shelf tool, gensim [23]. The average number of candidates per list is 41.

3.3 *Candidate Ranking*

In the candidate ranking phase, we formulate the ranking problem similar to [5, 24]. We generate a score function $f(m, e_i)$ based on features that extracted from the mention m and a candidate e_i. We select a candidate entity from candidate list E for a mention according to the highest score:

$$e = \text{argmax}_{e_i} f(m, e_i), e_i \in E \tag{1}$$

Therefore, the correct entry e for a mention m obtain a higher score than all other candidates $\hat{e} \in E, \hat{e} \neq e$. We use SVM$^{\text{rank}}$ [25] with the linear kernel to handle the optimization problem.

3.4 *NIL Classification*

We use heuristic rules to determine the final label for a mention. Mentions are labeled as NIL if there is no candidate in the candidate list or the ranking score of the top 1 candidate is below a threshold.

4 FEATURE STUDY

We extract multiple features for candidate ranking. First, we extract basic features from mention surfaces. In order to explore linguistic information in context, we categorize those linguistic-based features into several groups.

4.1 *Basic Features*

We focus on the surface properties of the KB title and the mention surface. The IsAcronym and IsAbbrMatch features [6] capture characteristics of acronyms. For example, given a mention *WTO*, acronym features can detect **W***orld* **T***rade* **O***rganization*. The SurfaceSimScore and EqualWordNumSurface features [26] calculate how similar is the mention surface to the KB title. The TokenLenInCandidate and CharLenInCandidate [12] count the terms and characters of the KB titles. We also incorporate other similarity features used in previous work [12, 27], such as dice coefficient scores and jaccard index scores. We summarize the basic features in Table 1.

4.2 *Linguistic Context Features*

We extract linguistic information from both mention source documents and texts of knowledge base entries (candidates) for disambiguation.

Title appearance Title appearance features [12] are related with the appearance of a candidate title in the source document, or the appearance of mentions in candidate texts. For example, if a given mention is the family name of a person, e.g., *Daughtry*, the title of a candidate, e.g., *Chris Daughtry*, may appear in the source document. Similarly, this given mention *Daughtry* may occur in the text of KB entry *Chris Daughtry*. Among them, a salient feature [12] detects disambiguators in candidate titles, e.g., *magazine* in People (magazine) and *basketball* in Maurice Williams (basketball).

Table 1. Basic Features of Candidate Ranking Module

Feature	**Description**
SurfaceSimScore	Levenshtein edit distance between the KB title and the mention surface
EqualWordNumSurface	Maximum of count of exact matches between mentions in the same group and the KB title
HasQueryGroup	Whether the KB title belongs to a mention group
QueryGroupMatch	Whether the KB title matches any surface in the same group
QueryGroupOverlap	Whether a surface in the same group is substring of the KB title, or vice versa
QueryGroupMaxSim	Maximum similarity between the KB title and surfaces in the same group
TokenLenInCandidate	Term count in the KB title
CharLenInCandidate	Characters count in the KB title
IsAcronym	Whether the mention surface is an acronym
IsAbbrMatch	Whether the capital character of the KB title match any surface in the same group
DiceTokenScore	Maximum value of the dice coefficient between the KB title token set and the surface token set
DiceToken	Whether DiceTokenScore is above 0.9
JaccardTokenScore	Maximum value of jaccard index between the KB title token set and the surface token set
DiceCharacterScore	Maximum value of dice coefficient between the KB title character set and the surface character set
DiceCharacter	Whether DiceCharacterScore is above 0.9
DiceAlignedTokenSocre	Maximum character dice coefficient of left and right aligned token sets
DiceAlignedToken	Whether DiceAlignedTokenScore is above 0.9
DiceAligned-CharacterSocre	Maximum character dice coefficient of left and right aligned character sets
DiceAlignedCharacter	Whether DiceAlignedCharacterSocre is above 0.9

Document similarity We use two measures to compare the text similarity between source documents and KB texts: cosine similarity with TF/IDF [26] and dice coefficient [27] on tokens. Since the first paragraphs of KB and text surrounding mention are supposed to be more informative, we consider using different ranges of source documents and KB texts. We divide text in a source document into local text (window size = 50 tokens) and global text (the whole source document), and use the first paragraph and the whole KB text receptively.

Entity mention occurrence Named entities in mention context are more salient than common words. This feature is used in [6], which could capture the count of co-occurring named entities between source documents and KB texts. For example, for a given mention *Obama*, the named entities *White House* and *United States* may appear in both the source document and the KB text if it refers to the American president *Barack Obama*.

Entity fact The infobox of KB contains important attributes of entries. For example, for entity *Apple Inc.*, we can extract attributes, such as Founder (*Steve Jobs*) and CEO (*Tim Cook*). Therefore we extract fact texts from KB and check whether fact texts are in source documents, which is inspired by [6]

Document topics Semantic information cannot be detected by simply counting occurrences of tokens, n-grams, and entities. Therefore we use topic models to discover the implicit topics of source documents and KB texts. We train LDA (Latent Dirichlet Allocation) model with gensim [23], which provides a fast online LDA model. We treat each KB entry as one document and use two different corpus for training. Zhang et al. [11] trained a topic model on the KBP knowledge base, we additionally train another topic model on the latest wikidump.[3] The KBP knowledge base is a partial KB and contains about one third of Wikipedia entities. We use two similarity measures to check the topic similarity between source documents and KB entries including cosine similarity and Hellinger distance. We also generate topics of partial text surrounding mention as the local topics to compare with using the whole source document (global topics).

Similarity of part-of-speech tokens We hypothesize that nouns and verbs compared to other type of words could contribute more on disambiguating. Therefore we collect this two type of tokens in context and calculate cosine similarity with TF/IDF weighting respectively.

Entity type We use entity type matching to detect whether the KB entity type is identical to the mention entity type, which is similar with [6]. For example, the mention *St.Andrew* is an ORG (Organization) entity in the first text in Figure 2. The candidate *University of St.Andrew* (ORG) is

[3] http://dumps.wikimedia.org/enwiki/20140707/enwiki20140707-pages-articles.xml.bz2

more likely than *St.Andrew, Scotland* (GPE) because of entity type matching. Therefore, we should predict named entity types for both non-NIL mentions and NIL mentions in the final output results. Since the KBP KB provides entity type information, we concern that it is more credible to predict non-NIL mention types by using KBP KB labeled type. However there are almost 64.9% unknown entities in the official KBP KB. It means that we need to re-tag remaining unknown entities. Unlike [6], we use the re-tag entity types according to our re-tagging results.

Clarke et al. [28] classified unknown type entities based on the infobox class. They also found that matching between infobox classes and entity types approximately has no ambiguity. Unlike [28] classified infobox class using learning method, we resolved around 2370 infobox classes manually. Our re-tagged result contains four types: PER, ORG, GPE, and MISC. Table 2 shows the comparison before and after retagging process.

Table 2. Entity types before and after re-tagging

Type	KBP KB	Our System
PER	14.0%	23.5%
ORG	6.8%	12.3%
GPE	14.2%	22.0%
UKN	64.9%	0.0%
MISC	0.0%	42.2%

5 Evaluation

5.1 *Data Set and Evaluation Metric*

We use the training data from the 2014 TAC KBP Entity Discovery and Linking (EDL) track [22]. The TAC data set consists of 5878 mentions over 158 documents. The statistics of the data set is shown in Table 3. We use the gold mention query file of the data set.

Our evaluation metric is micro-averaged accuracy, which is used in TAC KBP 2009 and 2010 entity linking task [13]. The metric is computed by

$$Accuracy_{micro} = \frac{NumCorrrect}{NumMentions} \tag{2}$$

Table 3. Statistics of 2014 TAC KBP Data set

	PER	ORG	GPE	Total
NIL	1819	591	216	2626
Non-NIL	1390	709	1153	3252
Total	3209	1300	1369	5878

5.2 *Experiment*

Since we focus on the ranking performance of each group of linguistic-based context features, we compute the accuracy of mentions system resolved. In order to eliminate the effect of feature combination, we add only one feature group to the basic feature group each time. We performed 5-fold cross-validation on the training set. Table 4 shows micro-averaged accuracies of feature addition experiments.

Table 4. Feature additive test results

Feature Group	**Non**	**NIL**	**ALL**
Basic	0.5910	0.7000	0.6394
Title Appearance	0.6138	0.7086	0.6558
Entity Fact	0.6024	0.6664	0.6306
Entity Mention Occurrence	0.6134	0.7668	0.6814
Document Similarity	0.6594	0.7733	0.7059
Document Similarity (LOCAL)	0.6422	0.7403	0.6860
Document Similarity (GLOBAL)	0.6474	0.7881	0.7096
Document Topic	0.6322	0.6912	0.6580
Document Topic (WIKI)	0.6224	0.6912	0.6528
Document Topic (KBP)	0.6280	0.6880	0.6544
Similarity of POS	0.6224	0.7420	0.6754
Similarity of POS (Noun)	0.6236	0.7364	0.6736
Similarity of POS (Verb)	0.5986	0.6970	0.6416
Type	0.5908	0.7030	0.6400
All Features	0.7330	0.7454	0.7378

In Figure 3, we consider subsets of mentions by entity type. We compare the entity linking performance on three types respectively.

In order to clarify feature effects, we divide features into more fine-grained groups, such as local topics (DT_WIKI_LOC, DT_KBP_LOC), global topics (DT_WIKI_GLO, DT_KBP_GLO), and document similarity by using the first paragraph of KB texts (DS_CON_FIR) or the whole KB

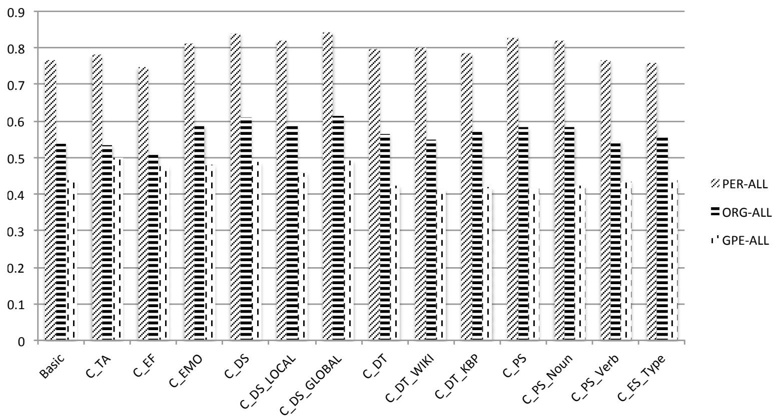

Fig. 3. Entity linking performance on PER, ORG, GPE entities

texts (DS_CON_ALL). Table 5 shows the increment of each fine-grained feature group to basic features on non-NIL mentions before NIL classification processing, and feature group names are capitalized referring to Table 4.

Moreover, we plan to check whether our current linguistic features can be used as local methods for global inference methods.Therefore, we compare the performance of our features with two local methods of previous work: 'TF/IDF cosine similarity' and 'Entity Popularity'. We evaluate the accuracy on the TAC KBP 2014 test data set. Since considering using local methods to filter out noisy candidates for global inference methods, we calculate the percentage of correct entities on different positions. The results are shown in Table 6. Acc@1, Acc@5, and Acc@10 mean correct entities on top1 position, within the top5 results, and within the top10 results.

We only focus on the performance of the in-KB entities because correct entities are provided in the data set. Similar with [20], we retrieve a mention with the Freebase search API,[4] and we select an entity which has the highest popular score of all the returned entities. We found that our features overcome 'TF/IDF cosine similarity' and 'Entity Popularity' in all the three cases.

[4] https://developers.google.com/freebase/v1

Table 5. Accuracy increment on non-NIL mentions before NIL classification

Fine-grained Feature Group	Accuracy Increment
C_DS_LOCAL	0.0736
C_DS_GLOBAL	0.1044
C_DS_CON_FIR	0.0214
C_DS_CON_ALL	0.0726
C_DT	0.0582
C_DT_WIKI	0.0338
C_DT_KBP	0.0576
C_DT_WIKI_GLO	0.0576
C_DT_WIKI_LOC	0.0344
C_DT_KBP_GLO	0.0534
C_DT_KBP_LOC	0.0510
C_PS_Noun	0.0658
C_PS_Verb	0.0150

Table 6. Accuracy at different positions

	Acc@1	Acc@5	Acc@10
TF/IDF cosine similarity	0.5066	0.8204	0.8910
Entity Popularity	0.3710	0.5453	0.6124
All Features	0.8264	0.9418	0.9492

6 DISCUSSION AND FUTURE WORK

6.1 *Overall Feature Effects*

Basic features only include features related to surface similarity, which is not effective enough to find correct entities. Features based on document similarity (both words and part-of-speech levels), named entities co-occurrence, and document topics contribute the most gains.

Document similarity In both document similarity and document topics, global features are better than local features. Since we leverage measures based on bag-of-words calculation, the larger text of context contains more co-occurring words than the window-size context. Although we suggest that the first paragraph in the KB is much informative, using the whole KB text (DS_CON_ALL) is much better than only using the first paragraph (DS_CON_FIR). We found that, in the KBP KB, several first paragraphs of KB texts are very short, sometimes only one sentence. For example, for *Jeff Perry (American actor)*, there is only one sentence, "Jeff Perry (born August 16, 1955 in Highland Park, Illinois) is an American character actor."

We found that around 28.74% entries of the KBP KB contain one simple sentence in the first paragraph.

Document topics Moreover, based on the results in Table 5, the increment of global topics is more than that of local topics by 0.024 (KBP corpus). Since the distribution of partial document topics is inconsistent with document topics, global topics can better represent the semantic context of a mention.

Although the KBP KB contains around one third entities of Wikipedia, the performance on the KBP KB corpus is better because we use the KBP KB as the entities database. We found that words of KBP KB topics could represent source document better than using the Wikipedia corpus for some entities. For example, *Salvador Dali* entity is a painter, who is also known for writing and film. Words of top topics are given by the KBP LDA corpus of this entity are *film*, *book*, *album*, *play*. However, words given by the Wikipedia LDA corpus are *Louisiana*, *disease*, *species*, and so on. The Wikipedia LDA corpus is not well-built, which may also affect the performance, because we follow an off-the-shelf training process.[5]

However, from Table 4 one can see that the performance on Wikipedia corpus is slightly effective on NIL by 0.003.

Similarity of POS tokens In Table 4, we found that nouns are more informative than verbs by around 0.3. Nouns contain more information than verbs because named entities are more salient.

6.2 *Feature Effects on Different Entity Types*

Figure 3 shows the performance on PER entities is much better than ORG and GPE entities by more than 20 percentage. It reveals that our features are biased toward PER entities and ORG and GPE mentions are difficult to resolve. After the error analysis, we found that simple string matching linguistic features fail to disambiguate ORG and GPE entities because of multiple name variation, especially the confusion between different entity types. For example, city names could be part of sport teams (*Orlando* is short for *Orlando Magic*) and people names could be part of company names (*Disney* is short for *Walt Disney Company* or *Walt Disney Animation Studio*). Moreover, the amount of PER mentions is two times larger than the amount of ORG mentions or GPE mentions in our data set.

[5] https://radimrehurek.com/gensim/wiki.html

6.3 *Future Work*

We compared the performance of current features with systems from the 2014 EDL Diagnostic task [22]. The accuracy on all mentions of our current system could beat the median system by 0.5, but we still have a huge gap with the best system. Even for some top systems from the 2014 EDL workshop [22] , performance on ORG and GPE entities are still much worse than PER entities. It should be an important future work to discover effective features which can solve ORG and GPE entities better.

Since we use a simple heuristic method to classify non-NIL and NIL mentions, the accuracy significantly drops after NIL classification process. In future work, we will explore effect features on determining NIL entities and improve the NIL classification method. Moreover, we plan to combine linguistic features with link-based methods to further improve our system.

REFERENCES

1. Ji, H., Grishman, R., Dang, H.T., Griffitt, K., Ellis, J.: Overview of the tac 2010 knowledge base population track. In: Third Text Analysis Conference (TAC 2010). (2010)
2. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to Wikipedia. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. (2011) 1375–1384
3. Erbs, N., Zesch, T., Gurevych, I.: Link discovery: A comprehensive analysis. In: 2011 Fifth IEEE International Conference on Semantic Computing (ICSC), IEEE (2011) 83–86
4. Hoffart, J., Yosef, M.A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G.: Robust disambiguation of named entities in text. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. (2011) 782–792
5. Bunescu, R.C., Pasca, M.: Using encyclopedic knowledge for named entity disambiguation. In: Proceedings of EACL. Volume 6. (2006) 9–16
6. Dredze, M., McNamee, P., Rao, D., Gerber, A., Finin, T.: Entity disambiguation for knowledge base population. In: Proceedings of the 23rd International Conference on Computational Linguistics. (2010) 277–285
7. Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: EMNLP-CoNLL. Volume 7. (2007) 708–716
8. Milne, D., Witten, I.H.: Learning to link with Wikipedia. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, ACM (2008) 509–518

9. Guo, Z., Barbosa, D.: Robust entity linking via random walks. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, ACM (2014) 499–508
10. Han, X., Sun, L., Zhao, J.: Collective entity linking in web text: A graph-based method. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM (2011) 765–774
11. Zhang, W., Sim, Y.C., Su, J., Tan, C.L.: Entity linking with effective acronym expansion, instance selection, and topic modeling. In: Proceedings of IJCAI. Volume 2011. (2011) 1909–1914
12. Graus, D., Kenter, T., Bron, M., Meij, E., De Rijke, M.: Context-based entity linking – University of Amsterdam at TAC 2012. In: Proceedings of the Fifth Text Analysis Conference (TAC 2012): November 5–6, 2012, National Institute of Standards and Technology, Gaithersburg, Maryland, USA. (2012)
13. McNamee, P., Dang, H.T.: Overview of the tac 2009 knowledge base population track. In: Text Analysis Conference (TAC). Volume 17. (2009) 111–113
14. Garcıa, N.F., Fisteus, J.A., Fernández, L.S.: Comparative evaluation of link-based approaches for candidate ranking in link-to-Wikipedia systems. Journal of Artificial Intelligence Research **49** (2014) 733–773
15. Han, X., Zhao, J.: Named entity disambiguation by leveraging Wikipedia semantic knowledge. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, ACM (2009) 215–224
16. Kulkarni, S., Singh, A., Ramakrishnan, G., Chakrabarti, S.: Collective annotation of Wikipedia entities in web text. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM (2009) 457–466
17. Guo, Y., Tang, G., Che, W., Liu, T., Li, S.: Hit approaches to entity linking at tac 2011. In: Proceedings of the Fourth Text Analysis Conference (TAC 2011). (2011)
18. Guo, S., Chang, M.W., Kiciman, E.: To link or not to link? a study on end-to-end tweet entity linking. In: HLT-NAACL. (2013) 1020–1030
19. Ferragina, P., Scaiella, U.: 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, ACM (2010) 1625–1628
20. Alhelbawy, A., Gaizauskas, R.: Graph ranking for collective named entity disambiguation. In: Proceedings of the 52nd Annual Meeting of the ACL. (2014) 75–80
21. Alhelbawy, A., Gaizauskas, R.: Collective named entity disambiguation using graph ranking and clique partitioning approaches. In: Proceedings of COLING. (2014) 1544–1555
22. Ji, H., Dang, H., Nothman, J., Hachey, B.: Overview of tac-kbp2014 entity discovery and linking tasks. In: Proceedings of Text Analysis Conference. (2014)

23. Řehůřek, R., Sojka, P.: Software framework for topic modelling with large corpora. In: Proceedings of the LREC: Workshop on New Challenges for NLP Frameworks. (2010) 45–50
24. McNamee, P., Dredze, M., Gerber, A., Garera, N., Finin, T., Mayfield, J., Piatko, C., Rao, D., Yarowsky, D., Dreyer, M.: Hltcoe approaches to knowledge base population at tac 2009. In: Proceedings of TAC 2009. (2009)
25. Joachims, T.: Training linear SVMs in linear time. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM (2006) 217–226
26. Zheng, Z., Li, F., Huang, M., Zhu, X.: Learning to link entities with knowledge base. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics (2010) 483–491
27. Dietz, L., Dalton, J.: Acrossdocument neighborhood expansion: Umass at tac kbp 2012 entity linking. In: Proceedings of TAC 2012. (2012)
28. Clarke, J., Suleiman, Y.M.G., Murgatroyd, S.Z.D.: Basis technology at tac 2012 entity linking. In: Proceedings of TAC 2012. (2012)

Shuangshuang Zhou
Inui-Okazaki Lab.,
Graduate School of Information Sciences,
6-6 Aramaki Aza Aoba, Aobaku, Sendai, Miyagi 980-8579,
Japan
E-mail: <shuang@ecei.tohoku.ac.jp>

Canasai Kruengkrai
Inui-Okazaki Lab.,
Graduate School of Information Sciences,
6-6 Aramaki Aza Aoba, Aobaku, Sendai, Miyagi 980-8579,
Japan
E-mail: <canasai@ecei.tohoku.ac.jp>

Naoaki Okazaki
Inui-Okazaki Lab.,
Graduate School of Information Sciences,
6-6 Aramaki Aza Aoba, Aobaku, Sendai, Miyagi 980-8579,
Japan
E-mail: <okazai@ecei.tohoku.ac.jp>

KENTARO INUI
INUI-OKAZAKI LAB.,
GRADUATE SCHOOL OF INFORMATION SCIENCES,
6-6 ARAMAKI AZA AOBA, AOBAKU, SENDAI, MIYAGI 980-8579,
JAPAN
E-MAIL: <INUI@ECEI.TOHOKU.AC.JP>