A Non-morphological Entity Boundary Detection Approach for Korean Text

Youngsik Kim[1], Younggyun Hahm[1], Dosam Hwang[2], Key-Sun Choi[3]

[1]KAIST, Korea, The Republic of
{twilight,hahmyg}@kaist.ac.kr
[2]Yeungnam University, Korea, The Republic of
dshwang@yu.ac.kr
[3]KAIST, Korea, The Republic of
kschoi@kaist.edu

Abstract. Even though being able to automatically annotate text with DBpedia URIs is a crucial step towards interconnecting the Web in any language, the problem of detecting and annotating DBpedia URIs within non-English text has not been widely tackled. In particular, no attempts have been made to automatically annotate DBpedia URIs within Korean text. Unlike previous Korean named entity recognition research, we tackle entity boundary detection as a separate problem. We describe an entity boundary detection approach for Korean text utilizing Support Vector Machines that does not require morphological annotations. We compare performance of this approach against several rule-based methods, including one that utilizes automatically annotated morphological annotations. Based on these results, we argue that several characteristics of the language makes entity boundary detection non-trivial enough that machine learning methods are preferable over rule-based methods, even with morphological annotations.

Keywords: Korean Entity Boundary Detection, Korean DBpedia, Text Annotation, Linked Data

1 Introduction

The ability to automatically annotate general text fragments with DBpedia URIs is an important step for connecting unstructured information within the Web of Data to structured Linked Data, and thus has been tackled in projects such as DBpedia Spotlight [1]. Although significant progress is being made for automatically annotating DBpedia URIs to English text, and research has been performed to extend DBpedia Spotlight to work with multiple languages [2], DBpedia Spotlight is not yet language-independent. Specifically, we have not found any previous research regarding the problem of automatically annotating DBpedia URIs to text in the Korean language.

The process of automatically annotating DBpedia URIs to text can be thought as having two steps: The first step, entity boundary detection, is to spot all possible entity candidates within the text; the second step, URI disambiguation, is to determine the correct DBpedia URI for each entity candidate given its context. The entity boundary detection step seems to be skimmed over in research targetting the English language.

adfa, p. 1, 2011.

For instance, entity boundary detection for DBpedia Spotlight is performed with a simple dictionary-based chunker [1]. While this is sufficient for some languages like English, simple chunking might not perform well or even be infeasible for languages where the average term length in the dictionary is very short.

In this paper we briefly explain why entity boundary detection for Korean text cannot be solved by rule-based chunking. Also, we evaluate the performance of several rule-based methods and Support Vector Machines, and demonstrate the possibility that separate machine training methods are required for satisfactory performance of the entity boundary detection step of automatically annotating DBpedia URIs to Korean text.

2 Related Work

Although we could not find previous work specifically about automatically annotating DBpedia URIs to Korean text, work about traditional Named Entity Recognition for Korean text do exist, such as utilizing Hidden Marcov Models [3] or a hybrid of statistical ML and rule-based algorithms [4]. Although these works both address the difficulty of detecting named entity boundaries in Korean text compared to that in English, no separation was made between the boundary detection and the classification steps. More importantly, both works do not distinguish between boundary detection errors and classification errors. Although the scope of entities of our research (Korean DBpedia entities) differs from that of these works (Traditional named entity classes such as DATE, PERSON, ORGANIZATION), we believe that in both cases boundary detection is non-trivial enough that performance will benefit by separately considering boundary detection and classification.

Research about a language-independent model for the Entity Detection and Tracking task has also been previously performed, using Arabic, Chinese, and English texts for evaluation [5].

3 Problem Definition

3.1 The Dataset

We used the entire 2014/01/26 Korean Wikipedia dump as the source dump of our dataset. We used Wikipedia Extractor[1] to extract plain text and link information of each Wikipedia article from the source dump. While this program is most likely intended for parsing Italian Wikipedia dumps, we found that it works as well for parsing the Korean Wikipedia dump. We assumed that the performace in terms of preservability of plain text and links was adequate for our purposes.

Our dataset consists of the set of all Korean Wikipedia articles $Article_1, Article_2, \cdots,$ Article$_N$ where each article $Article_x$ consists of:

- $Title_x$, the title of $Article_x$.

[1] http://medialab.di.unipi.it/wiki/Wikipedia_Extractor

- $Text_x$, the text of $Article_x$, which is a list of consecutive characters $c_{x1}, c_{x2}, \cdots, c_{x\ len(text_x)}$.
- A list of existing links $Link_{x1}, Link_{x2}, \cdots, Link_{xL}$, where $Article_x$ has L existing links. Each link $Link_{xl}$ in turn consists of:
  - The positions $LinkStart_{xl}$ and $LinkEnd_{xl}$ representing the start and end character offsets of the link. The surface form $Surface_{xl}$ of this link is equivalent to the characters $c_{x\ LinkStart_{xl}} \sim c_{x\ LinkEnd_{xl}}$.
  - $LinkURI_{xl}$, the destination URI of the link. We only consider links that point to other Korean Wikipedia articles.

3.2 The Scope of Entities

We limited the scope of entities to our Korean Wikipedia dataset, meaning that all and only existing Korean Wikipedia article URIs are considered to be entities. Furthermore, we made two extra considerations:

1. Redirection: Some Wikipedia URIs redirect to other Wikipedia URIs. In this case, we always used the destination URI as the actual entity for referrer URIs. No circular redirections existed within the dataset.
2. Disambiguation: Some Wikipedia URIs only serve as a 'medium' article for multiple URIs that share the same written form. Since these URIs describe no actual entities by themselves, we discarded all such URIs from the set of entities.

We defined the entity scope to be the set of all Korean Wikipedia article URIs that are not disambiguation articles. Due to this definition, existing links that point to disambiguation articles were ignored.

3.3 The Scope of Entity Boundary Detection

We built a dictionary consisting of surface forms of every link within our dataset, and limited the domain of entity boundary detection to sequences of characters within the text that are members of the dictionary. More formally, we defined the dictionary SurfaceDict as:

$$\text{SurfaceDict} = \{C | \exists x \exists l (C \equiv Surface_{xl})\}$$

and the domain $Candidates_x$ of possible entity boundary offsets for $Text_x$ as:

$$Candidates_x = \{(start, end) | c_{x\ start} \sim c_{x\ end} \in SurfaceDict\}$$

Moreover, since entities within the same text must not overlap with each other, any possible ValidBoundaries_x produced by entity boundary detection for $Text_x$ must be a subset of Candidates_x and contain no entities whose boundaries overlap.

Since the Korean language offers no obvious clues (such as capitalization) for entity boundary detection, the task of determining whether an arbitrary sequence of characters represents an entity or not without a dictionary is outside the scope of this paper.

4 Experiments

4.1 The Answer Set

We assumed the entirety of the set of existing links to be accurate: Both the boundaries and assigned entity URI of every existing link are to be trusted. Of course, this does not mean the set of existing links in an article is equal to the set of entities in the article; there are typically many more entities within a single text that are not links than entities that are links. Brief observations on the dataset suggested that even surface forms with a 1:1000 linked entity to non-linked entity ratio could still be considered to be an entity most of the time.

Because of this, manual annotation was required to create a golden standard of articles annotated with entity information. Manual annotation for a single article was performed by comparing the clean text of the article with the two dictionaries SurfaceDict and SemanticMeaning, following a set of guidelines directly derived from the MUC-7 Named Entity Task Definition guidelines [6]. The answer set of an article $Article_x$, $AnswerLinks_x$, would be a complete set of links within the article so that $AnswerLinks_x \supset \{Link_{x1}, Link_{x2}, \cdots, Link_{xL}\}$.

All experiments were performed with a golden standard created by 3 annotators consisting of 55 different Korean Wikipedia articles, each article having 20 to 50 sentences and a (0.5 ~ 4):1 existing link to sentence ratio. These conditions were enforced to prevent 'strange' articles (single-sentence articles, articles consisting of lists of names…) from skewing the experiment results. The union of $Candidates_x$ for all 55 articles had 9416 entities and 90221 non-entities.

4.2 Outline of the Entity Boundary Detection Algorithm Process

Because many Korean surface forms exist that are only 1 or 2 characters long, there is an abundance of overlapping surface forms within $Candidates_x$ for any typical $Text_x$. This is shown in Figure 1, where basically all characters within the sentence are entity candidates on their own. The approximately 1:10 entity to non-entity ratio observed in our answer set suggests that this is typical of Korean text in general.

Fig. 1. A sample sentence "Gyeongjong of Goryeo was the fifth ruler of the Goryeo dynasty of Korea." annotated with boundaries within its $Candidates$. Each bracket represents a single candidate surface form, and the bold brackets represent the answer set for this sentence.

The obvious solution would be to perform chunking on the text beforehand, and deciding whether each chunk is an entity or not. Unfortunately, chunking in Korean is not trivial due to the fact that both postpositions and compound nouns are extremely

commonplace in the language. The chunking performance of the current state-of-the-art Korean morpheme tagger was unsatisfactory: More than 10% of the existing links within the dataset were unrecoverable using the chunks generated by the tagger because the boundaries of chunks did not align with the boundaries of the links.

Thus, we decided to also consider an alternative process that does not rely on a single chunking scheme. For each sentence, the process first determines whether each candidate surface form in the whole $Candidates$ set of the sentence is a valid surface form or not, using an entity candidate subset selection method described in Section 4.3.

The process must then refine this subset into a set that contains no surface forms with overlapping boundaries. Since DBpedia entities are nouns, the majority (if not all) of overlapping valid surface forms in Korean text occur in compound nouns where the boundary of one surface form completely contains the others. Thus, we decided to resolve overlapping candidate surface forms by greedily selecting the ones with the longest boundary length while scanning the candidates in the order they appear within the sentence. An example visualization of this process is shown in Figure 2.

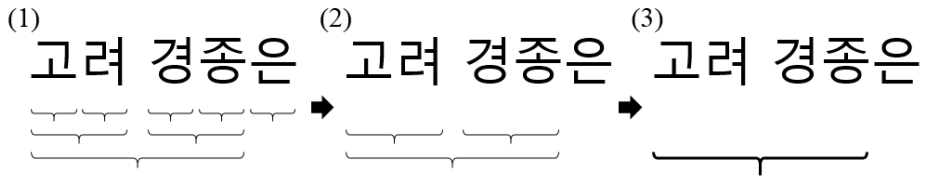

Fig. 2. An example of the process described in Section 4.2, upon the phrase "Gyeongjong of Goryeo was". Starting with the whole $Candidates$ set (1), the process determines the validity of each candidate surface form to produce a subset of surface forms (2). Overlapping surface forms are resolved by choosing the ones with the longest boundary length (3).

We now present multiple entity candidate subset selection methods, some rule-based and others using Support Vector Machines with various kernels.

4.3 Entity Candidate Subset Selection Methods

Baseline. The baseline method does not perform any subset selection, and uses the entire set of $Candidates$.

Word-based. The word-based method selects the subset of candidates that consists of all surface forms within the text that satisfies the following conditions: Both the character immediately before the surface form and the character immediately after the surface form are either whitespace or other special characters. This is meant to simulate the performance of word-based entity boundary detection methods used for English text.

Suffix-based. The suffix-based method is an extension of the word-based method in which the character(s) following the surface form that are part of the same (last) word as the surface form are either white space, special characters, or match any of the 27 different pre-defined strings of letters. No morphological analysis is required for this method.

The basis for this method came from statistical analysis of existing links within the dataset: >99% of the links were prefixed with whitespace or a special character, and >95% of the links were suffixed with whitespace, a special character, or with one out of 27 different strings of letters which all are common Korean postpositions.

Morphology-based. The morphology-based method selects the subset of candidates that consists entirely of noun-like morphemes ('nc', 'np', 'nn', 'nb') and nominal suffixes ('xsn'), based on tagged text generated by a state-of-the-art Korean morpheme tagger. This rule covers >95% of the existing links in the dataset, exculding the links that could not be retrieved due to chunking boundary alignment errors.

Note that this method essentially produces a non-overlapping sequence of chunks, and is identical to the chunking method briefly mentioned in Section 4.2.

SVM-based. This method uses Support Vector Machines with various kernels to determine a subset of candidates. Since features used for SVM training cannot be strings[2], we used conditional probabilities involving the surface form and nearby characters as features.

If we define two functions, $\mathrm{C}(s)$: The surface form s satisfies the condition C(in which the specific condition defines each particular feature) and $\mathrm{E}(s)$: The surface form s is an entity, the conditional probability can be calculated as $\mathrm{P}(E(s)|C(s))$. But since the current golden standard is small, the features generated by using the golden standard would be very sparse. Thus, we had to use the entire dataset and the set of existing links to generate denser feature values.

In order to do so, we made the following assumption: If $\mathrm{Link}(s)$ is defined as "The surface form s is tagged as an existing link", the conditional probability $\mathrm{P}(Link(s)|E(s))$ is mostly constant and does not depend on the surface form s. In doing so, we substitute the probability $\mathrm{P}(Link(s)|C(s))$ for $\mathrm{P}(E(s)|C(s))$ for every feature.

We used a total of 7 features, which all are based on the surface form and its surrounding characters. The condition $\mathrm{C}(s)$ of each feature for a surface form *T* are: *T* itself, the prefix/suffix of *T*, the word immediately before/after *T*, and combinations of the previous conditions. All features are letter-based and do not require morphological analysis beforehand.

All of the SVM variations were binary classifiers run with the scikit-learn framework [7], using the same set of features with differing kernels. All parameters not listed for each variation were assigned their respective defaults.

- SVM-1: Linear kernel, identical weights for both classes.
- SVM-2: Linear kernel, automatically weighted classes.
- SVM-3: 3-degree polynomial kernel, identical weights for both classes.
- SVM-4: 3-degree polynomial kernel, 3:1 weights for the entity:non-entity classes.
- SVM-5: 3-degree polynomial kernel, 6:1 weights for the entity:non-entity classes.
- SVM-6: 3-degree polynomial kernel, automatically weighted classes.
- SVM-7: RBF kernel, identical weights for both classes.

[2] It is possible to use the Levenshtein distance between strings to create a string-based kernel, but this kernel obviously is inappropriate for our problem because similar-looking surface forms do not have similar probabilities of being entities.

- SVM-8: RBF kernel, automatically weighted classes.

4.4 Performance Evaluation

We followed the same performance evaluation metrics as described in the CoNLL-2003 shared task [8]. Each method was evaluated using 5-fold cross-validation, with identical splits for each method.

5 Results

Method	Precision	Recall	F-score
Baseline	21.03	**92.85**	34.28
Word-based	50.55	35.54	41.69
Suffix-based	62.70	78.71	69.74
Morphology-based	55.64	91.20	69.11
SVM-1	**92.03**	63.13	74.86
SVM-2	65.59	89.03	75.52
SVM-3	88.06	72.24	79.33
SVM-4	76.83	85.26	**80.81**
SVM-5	69.75	88.58	78.03
SVM-6	66.43	89.21	76.14
SVM-7	86.45	74.36	79.92
SVM-8	64.95	90.57	75.65

Table 1. Performance metrics of each method.

As shown in Table 1, the SVM-based methods generally performed better than the rule-based methods. Based on the fact that the Suffix-based and Morphology-based methods show roughly the same F-score performance, we propose that the performance of rule-based methods have an upper bound for Korean entity boundary detection that can be improved upon with machine learning algorithms such as SVM.

In particular, the fact that the recall of the Morphology-based method which used a state-of-art Korean tagger is not much higher than the recall of the SVM-based methods shows that Korean morphological taggers in their current state are not required in entity boundary detection of Korean text.

Out of the SVM-based methods, SVM-4 (3-degree polynomial kernel, 3:1 weights for the positive:negative classes) reported the best F-score, but the differences with other SVM-based methods are mostly statistically insignificant.

The high recall of our baseline shows that our decision to resolve overlapping surface form candidates by selecting the longest ones mostly holds. Some exceptions, such as when a valid surface form combined with its suffix forms a longer (but invalid) candidate surface form, do exist but are relatively uncommon.

6 Discussion and Future Work

In this paper, we have only covered the entity boundary detection step of annotating DBpedia URIs to Korean text. As such, the best-performing method described here might not be the best method to use in the task of automatic annotation as a whole. For example, setting a lower bound on confidence scores for URI disambiguation in order to filter out non-entities is one possible alternate method to the two-step DBpedia URI annotation process.

After the problem of URI disambiguation for Korean candidates has been tackled, we are currently planning to release the entire DBpedia URI annotation pipeline as a Web service. This plan requires consideration of memory requirements and running time, both of which we have not analyzed yet.

7 References

1. Mendes, Pablo N., et al. "DBpedia spotlight: shedding light on the web of documents." *Proceedings of the 7th International Conference on Semantic Systems*. ACM, 2011.
2. Daiber, Joachim, et al. "Improving efficiency and accuracy in multilingual entity extraction." *Proceedings of the 9th International Conference on Semantic Systems*. ACM, 2013.
3. Chung, Euisok, Yi-Gyu Hwang, and Myung-Gil Jang. "Korean named entity recognition using HMM and CoTraining model." *Proceedings of the sixth international workshop on Information retrieval with Asian languages-Volume 11*. Association for Computational Linguistics, 2003.
4. Seon, Choong-Nyoung, et al. "Lightweight Named Entity Extraction for Korean Short Message Service Text." *KSII Transactions on Internet and Information Systems (TIIS)* 5.3 (2011): 560-574.
5. Florian, Radu, et al. *A statistical model for multilingual entity detection and tracking*. IBM THOMAS J WATSON RESEARCH CENTER YORKTOWN HEIGHTS NY, 2004.
6. Chinchor, Nancy, and Patricia Robinson. "MUC-7 named entity task definition."*Proceedings of the 7th Conference on Message Understanding*. 1997.
7. Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.
8. Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: language-independent named entity recognition. In Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4 (CONLL '03), Vol. 4. Association for Computational Linguistics, Stroudsburg, PA, USA, 142-147. DOI=10.3115/1119176.1119195 http://dx.doi.org/10.3115/1119176.1119195

8 Acknowledgements

This work was supported by the ICT R&D program of MSIP/IITP. [10035348, Development of a Cognitive Planning and Learning Model for Mobile Platforms] [10044494, WiseKB: Big data based self-evolving knowledge base and reasoning platform]