LETTER

Utilizing the Web for Automatic Word Spacing

Gumwon HONG[†], **Jeong-Hoon LEE**[†], **Young-In SONG**[†], **Do-Gil LEE**[†], ***Nonmembers***, ***and*** **Hae-Chang RIM**[†*a)], ***Member***

SUMMARY This paper presents a new approach to word spacing problems by mining reliable words from the Web and use them as additional resources. Conventional approaches to automatic word spacing use noise-free data to train parameters for word spacing models. However, the insufficiency and irrelevancy of training examples is always the main bottleneck associated with automatic word spacing. To mitigate the data-sparseness problem, this paper proposes an algorithm to discover reliable words on the Web to expand the vocabularies and a model to utilize the words as additional resources. The proposed approach is very simple and practical to adapt to new domains. Experimental results show that the proposed approach achieves better performance compared to the conventional word spacing approaches.

key words: *word spacing, word segmentation*

1. Introduction

Word spacing is the task of finding correct word[**] boundaries in Korean text. The automatic word spacing task for Korean can be regarded as a general word segmentation task for Chinese or Japanese; when spacing a Korean text, all the existing spaces are first removed and the text is segmented into words by finding the correct positions to insert spaces.

Conventional approaches to statistical word spacing tasks train models on noise-free corpora; gathered from reliable sources, they are usually composed of well written texts such as news articles and literature from various genres. Based on syllable[***] n-gram statistics, the statistical models basically compute the probability that given two syllables should be combined or separated. Unlike Chinese or Japanese, there are many sources for Korean word spacing to acquire well-spaced texts such as online news articles, and often times, training on them guarantees reasonable performances; many systems reported more than 96% of syllable-unit accuracy when tested on literary style texts [1]–[4].

Though the approaches seem reasonable, in practical situations they have some limitations. First, collecting training data in a target domain is sometimes difficult. When correcting spacing errors in SMS messages, training on corrected SMS messages are preferred. Data set construction is however a costly and time consuming task. Second, due to unknown words caused by the lexical discrepancy between training data and testing data, the conventional approaches are not effective when tested on colloquial texts. New words, such as jargons, abbreviations or newly-coined words, are continuously generated in SMS messages, forums, blogs or comments on the Web, which produces frequent out-of-vocabulary (OOV) words. For example, Sejong corpus (see Sect. 5.1) contains no such words as *ji-mot-mi*[****] or *lo-tto*. Spacing decision between *ji* and *mot* (or *lo* and *tto*) is made by depending only on a few surrounding syllables, and false spacing insertion is generated between *ji* and *mot* (or *lo* and *tto*) because of extremely high probability that two given syllables should be split.

As a solution to this data-sparseness problem, one might expect to use the Web as training corpus. Because of its heterogeneous nature, the Web incorporates numerous words from various domains. Currently, many researchers address the issue regarding the Web as corpus to resolve the data-sparseness problem in NLP [5]–[7]. However, due to the lack of editorial process, many user generated texts on the Web contain lots of spacing errors. For example, around 60% of the sentences in the corpus collected from blogs, forums, and comments contain at least a spacing error. Because of these errors on the Web, without devising a suitable method of utilizing reliable information from the Web, directly using the Web texts may not be effective in enhancing the performance of automatic word spacing.

2. Observation

The spacing errors are classified into two categories: spacing omission error and spacing insertion error. When analyzing the spacing errors, we found that spacing omission errors are more likely to be made than spacing insertion errors. As will be shown in Table 2 of Sect. 5.1, the omission error ratio in our Web corpus is about 87%.

Figure 1 exemplifies the types of errors: the first line shows the sequence of four correct words which are denoted as $w_1 \ldots w_4$, and the second and the third lines contain an omission error and an insertion error, respectively. In the

Manuscript received November 19, 2008.
Manuscript revised May 5, 2009.
[†]The authors are with Korea University, Korea.
[*]Corresponding author.
a) E-mail: rim@nlp.korea.ac.kr
DOI: 10.1587/transinf.E92.D.2553

[**]Precisely, Korean spacing unit is Eojeol which is possibly composed of one or more words. We simply use the term *word* to denote *Eojeol* for convenience.

[***]A syllable denotes a Korean character which is composed of two or more letters.

[****]An abbreviation representing "I'm sorry for not protecting you."

	w_1	w_2	w_3	w_4
1.	*sa-yong-hal*	*su*	*it-neun*	*bang-beop*
2.	*sa-yong-hal-su*	*it-neun-bang-beop*		
3.	*sa-yong-hal-su*	*it*	*neun*	*bang-beop*

Fig. 1 Example of errors on the Web.

second line, one space between w_1 and w_2 and another between w_3 and w_4 are omitted, which produces two multi-words w_1w_2 and w_3w_4. The third line shows that a false space is inserted between *it* and *neun*. However, the probability of observing the last example is very low on the Web corpus. This observation leads us to assume that texts on the Web can be regarded as a set of chunks, each of which consists of one or more correct words.

We found that if a chunk occurs frequently, then it can be regarded as a correct word, whereas an infrequently occurring one often contains many omission errors, hence it must be further segmented into correct words. Therefore, we can select words whose frequencies are above a certain threshold as reliable words and use them as additional resources for automatic word spacing.

Based on this observation, we propose an approach which mines reliable words from the Web and use them as additional resources for automatic word spacing. The remaining sections of this paper will focus on the word mining algorithm to discover reliable words and the model to utilize the mined reliable words.

3. Word Mining Algorithm

To find reliable words from the Web corpus, we propose a word mining algorithm presented in Fig. 2. The algorithm iteratively estimates word probabilities and applies them to each chunk in a corpus to generate a sequence of short correct words.

We discuss the algorithm formally with some notations. Let us assume that an initial set $X^{(0)}$ of words are extracted from raw corpus and a frequency f_x for each $x \in X^{(0)}$ is greater than zero. At each iteration t, a new set $X^{(t+1)}$ of words is constructed from the current set $X^{(t)}$ of words in the following self-training fashion. The set $X^{(t)}$ is divided, based on the reliability of a word which is measured by a frequency threshold θ, into two disjoint sub sets: a set $L^{(t)}$ of correct words and a set $U^{(t)}$ of erroneous words. In this study, we set θ as 2 empirically. We train a word model $M^{(t)}$ (which will be discussed in Sect. 4) on $L^{(t)}$ set, and apply the model to segment each chunk x in $U^{(t)}$ set. If the chunk is segmented into two or more words, then it may consist of known words and new words. While, the algorithm only updates the frequency of a known word w_k by adding the frequency of the chunk x to f_{w_k}, it adds a new word w_n to $X^{(t+1)}$ and sets the frequency as f_x. If the chunk is not segmented, the algorithm just stores the chunk in $X^{(t+1)}$ by leaving the segmentation in the later iteration. The algorithm stops when the set of words and their frequencies do not change.

Initialization
$X^{(0)}$: Set of words extracted from raw corpus
f_x: A frequency for each word $x \in X^{(0)}$

Iteration
for $t \in \{0, 1, ...\}$
  Divide $X^{(t)}$ into $L^{(t)}$ and $U^{(t)}$, where
  $L^{(t)} = \{x \in X^{(t)} | f_x > \theta\}$ and $U^{(t)} = \{x \in X^{(t)} | f_x \leq \theta\}$
  Set $X^{(t+1)} = L^{(t)}$
  Train a word model $M^{(t)}$ on $L^{(t)}$
  For each chunk $x \in U^{(t)}$:
    Segment x into words $\hat{W} = w_1, ..., w_m$ based on $M^{(t)}$
    If $m > 1$ then
      For each known word $w_k \in X^{(t)}$ in $\hat{W}$
        Set $f_{w_k} = f_{w_k} + f_x$
      For each new word $w_n \notin X^{(t)}$ in $\hat{W}$
        Set $X^{(t+1)} = X^{(t+1)} \bigcup \{w_n\}$
        Set $f_{w_n} = f_x$
    Otherwise,
      Set $X^{(t+1)} = X^{(t+1)} \bigcup \{x\}$
  If $X^{(t)} = X^{(t+1)}$, stop

Fig. 2 Word mining algorithm from Web corpus.

4. A Model Incorporating Word and Syllable Information

We propose a stochastic word spacing model which uses the Web as additional training data. The model heuristically combines syllable information extracted from noise-free data and word information discovered on the Web using the word mining algorithm. Syllable information is used to estimate the probability that given two syllables would be separated. Word information supplements the word spacing decision by estimating the probability that a sequence of syllables would form a word.

Given a syllable sequence $C = c_1 \dots c_m$, the proposed model tries to find an optimal word sequence $\hat{W} = \hat{w}_0 \dots \hat{w}_n$ out of all possible segmentations W as:

$$\hat{W} = \underset{W}{\operatorname{argmax}}\, \lambda \times \sum_{i=1}^{n} \log(P(w_i|s_{i-1}, c_{b_i,e_i}, s_i)) + (1-\lambda) \times \sum_{j=0}^{n} \log(P(s_j|c_{l_j}, c_{r_j})), \quad (1)$$

where $P(w_i|s_{i-1}, c_{b_i,e_i}, s_i)$ estimates the likelihood that a sequence of syllables c_{b_i,e_i} forms a word w_i, and $P(s_j|c_{l_j}, c_{r_j})$ measures the likelihood that a space s_j is located between the left context c_{l_j} and the right context c_{r_j}. If candidate spaces s_{i-1} and s_i are selected based on syllable information (i.e., surrounding contexts), then the model supplements the decision by examining whether the syllables between the two spaces actually form a word or not.

We denote $P(s_j|c_{l_j}, c_{r_j})$ a *spacing probability* and $P(w_i|s_{i-1}, c_{b_i,e_i}, s_i)$ a *word probability*. The spacing probability is basically the same as the probability used by Kang et al. (2001) as in [2]. However, we allow more context than Kang et al. (2001) in estimating the probability; we look at two syllables to the left and two syllables to the right from the position where a space should be inserted, and we use the

smoothing method introduced in Thorsten Brant's algorithm as in [8]. The *word probability* is used as a *word model* in our algorithm described in the previous section. To estimate the word probability, we use following equation:

$$\hat{P}(w_i|s_{i-1}, c_{b_i,e_i}, s_i) \approx \frac{freq(c_{b_i,e_i} = w_i) + 1}{freq(c_{b_i,e_i}) + 2^{|w_i|+1}}, \quad (2)$$

where the denominator is basically the frequency of a syllable sequence $c_{j,k}$, and the numerator is the number of times that the sequence form a word w_i. Because the probability of a sentence is defined by the sum of individual logs of the probabilities, the model inherently prefers fewer segments. To mitigate this problem, we normalize the word probability by adding $2^{|w_i|+1}$, the number of possible segmentations for a syllable sequence of length $|w_i|$, to the denominator, and by adding 1 to the numerator. Thus, in case of $freq(c_{b_i,e_i}) = 0$, the equation estimates a likelihood that a sequence of syllables constitute a word by chance. To find the best segmentation among possible candidates, we use the Viterbi algorithm as in [9].

5. Experiments

5.1 Experimental Setup

The purpose of our experiment is to verify the effectiveness of the proposed approach for correcting spacing errors in colloquial texts. We choose two target domains for our experiments, SMS and Web, where spacing errors are prevalent. On each target domain, we compare the proposed approach, which discovers reliable words from the Web and utilizes them as additional resources, with other approaches.

Data sets Table 1 presents the collections for the experiment. Our data sets consist of

- *literary*: a noise-free corpus which is the combination of Sejong corpus[†] and online news articles. This corpus covers various genres of literary-style texts.
- *web*: a collection of texts consisting of forums, blogs and comments which are crawled from 20 popular web sites.
- *sms*: a set of text messages typed and sent via mobile phones. Due to the limitation of bytes to be sent at a time, it contains frequent spacing omissions errors.

Manual correction of word spacing errors for two test sets, *sms* corpus and *web* corpus, is performed by the five graduate students. We randomly extracted 5,000 sentences from *web* corpus for testing. Table 2 shows spacing accuracies and the omission error rate in our test data. *sms* corpus is the noisiest data, mainly consisting of colloquial words, whereas *web* corpus contains both colloquial and literary-style words, hence it is less noisy than *sms*.

Table 1 Training & testing data statistics.

	training		testing	
domain	*literary*	*web*	*sms*	*web*
#words	7.7M	410M	315,095	30,927

Evaluation measure We used three measures for evaluation: syllable-unit accuracy, sentence-unit accuracy and word-unit recall.

$$A_{syl} = \frac{\#\ of\ correctly\ spaced\ syllables}{\#\ of\ syllables\ in\ test\ data}$$

$$A_{sen} = \frac{\#\ of\ correctly\ spaced\ sentences}{\#\ of\ sentences}$$

$$R_{wrd} = \frac{\#\ of\ correctly\ spaced\ words}{\#\ of\ words\ in\ test\ data}$$

5.2 Experimental Results

The effectiveness of the proposed approach: The first experiment is to show the effectiveness of the approach which uses reliable words generated by the word mining algorithm on the Web as additional resources. Table 3 shows the comparative experimental results of three approaches. All three approaches are based on the model described in Sect. 4, but they are different from each other in the way that they learn the model parameters.

- *Baseline 1*: This approach estimates model parameters only on a noise-free corpus. Both the syllable probabilities and the word probabilities are trained on *literary* corpus in this approach.
- *Baseline 2*: This approach trains model parameters on combined noise-free corpus and Web corpus. The word mining algorithm is not used in this approach; rather, the syllable probabilities and word probabilities are directly trained on the *literary* and *web* corpora.
- *Proposed*: This is the proposed approach which trains syllable probabilities on *literary* corpus and word probabilities via the word mining algorithm on *web* corpus.

A large improvement from the two baseline approaches shows the effectiveness of the proposed approach. The performance difference is much greater in word-unit recall which supports our hypothesis that the proposed approach can mitigate the data-sparseness problem from the case when only noise-free training data are used. Interestingly, baseline 2 slightly increases the accuracies in SMS domain. However, this is not the case in Web domain. Moreover, the performance improvement of the approach is only

Table 2 Spacing accuracies and omission error rate ($oer = omission/(omission + insertion)$) before error correction.

sms				*web*			
A_{syl}	A_{sen}	R_{wrd}	*oer*	A_{syl}	A_{sen}	R_{wrd}	*oer*
74.4	13.9	15.3	98.8	92.9	39.5	65.4	86.5

Table 3 The effectiveness of the proposed approach.

test set	sms			web		
	A_{syl}	A_{sen}	R_{wrd}	A_{syl}	A_{sen}	R_{wrd}
baseline 1	88.9	47.5	61.2	93.7	38.3	72.0
baseline 2	89.6	52.4	66.3	93.1	36.2	69.4
proposed	**91.6**	**56.4**	**73.3**	**95.3**	**51.0**	**82.2**

[†]See the 21st Century Sejong Project's home page at http://www.sejong.or.kr/english.php.

Table 4 Comparative results with conventional approaches.

test set	sms			web		
	A_{syl}	A_{sen}	R_{wrd}	A_{syl}	A_{sen}	R_{wrd}
HMM	90.4	52.6	63.8	93.7	36.1	66.8
CRF	89.7	52.1	65.4	92.6	35.9	68.7
proposed	**91.6**	**56.4**	**73.3**	**95.3**	**51.0**	**82.2**

marginal when compared to the proposed approach.

Comparison with conventional approaches: In this experiment, we try to compare the performance of the proposed approach with previous state-of-the art approaches. Table 4 compares the results of the proposed approach with the two conventional approaches: *HMM* and *CRF*.

- *HMM*: the approach based on the HMM model, which is known to be the state-of-the art in generative word spacing models, introduced in [4]. They regard the word spacing problem as the tagging problem; every syllable has either tag "1" or tag "0": the former means a syllable is followed by a space whereas the other means it is not followed by a space.
- *CRF*: the approach which uses CRF model, which has shown good performance in Chinese word segmentation tasks [10], [11]. We use linear-chain CRFs[†] and the standard BIO labels. In this experiment, we use the feature set which was introduced in [10].

Like the baseline approaches, the two conventional approaches do not perform well on the colloquial test sets. On the other hand, the proposed approach significantly improves the performance: the average improvements in word-unit recall for SMS and Web domains are 13.5% and 21.4%, respectively.

6. Conclusions

In this paper, we presented a novel approach which mines reliable words from the Web and use them as additional resources for automatic word spacing. The lexical discrepancy between the literary-style training data and the colloquial testing data causes a serious data-sparseness problem in automatic word spacing.

The main contributions of this paper are the following: 1) we devise an algorithm that finds reliable words on the Web, 2) we propose a model to combine the word information mined from the Web and syllable information extracted from noise-free corpus and 3) consequently, we can devise a method of alleviating the data-sparseness problem in automatic word spacing.

When evaluated on two different target domains, the proposed approach achieved better performance, in all measures, compared to the conventional approaches. This result proves that the mined words from the Web can be used as useful resources for automatic word spacing.

Though the proposed approach can directly approximate the word probability on the Web, it still relies on syllable statistics of noise-free data to estimate the spacing probability. For future work, we plan to devise an approach to mining the Web for reliable spacing information as well as word information.

[†]We used Taku Kudo's CRF++ package version 0.51. Refer to http://crfpp.sourceforge.net for more information.

Acknowledgments

This research was supported by Grant R01-2006-000-11162-0 (2008) from the Basic Research Program of the Korea Science & Engineering Foundation.

References

[1] K. Shim, "Automated word-segmentation for Korean using mutual information of syllables," J. Korea Information Science Society, pp.991–1000, 1996.
[2] S.S. Kang and C.W. Woo, "Automatic segmentation of words using syllable bigram statistics," Proc. NLPRS2001, pp.729–732, 2001.
[3] S.B. Park, Y.S. Tae, and S.Y. Park, "Self-organizing n-gram model for automatic word spacing," Proc. ACL 2006, pp.633–640, ACL, 2006.
[4] D.G. Lee, H.C. Rim, and D. Yook, "Automatic word spacing using probabilistic models based on character *n*-gram," IEEE Intelligent Systems, vol.22, no.1, pp.28–35, Jan. 2007.
[5] A. Kilgarriff and G. Greffenstette, "Introduction to the special issue on web as corpus," Computational Linguistics, vol.29, no.3, pp.1–15, 2003.
[6] V. Liu and J.R. Curran, "Web text corpus for natural language processing," EACL, 2006.
[7] F. Keller and M. Lapata, "Using the web to obtain frequencies for unseen bigrams," Computational Linguistics, vol.29, no.3, pp.459–484, 2003.
[8] T. Brants, "TNT—A statistical part-of-speech tagger," Proc. 6th Applied Natural Language Processing Conf. (ANLP 00), pp.224–231, 2000.
[9] A.J. Viterbi, "Error bounds for convolutional codes and an asymmetrically optimum decoding algorithm," IEEE Trans. Inf. Theory, vol.13, no.2, pp.260–269, 1967.
[10] F. Peng, F. Feng, and A. McCallum, "Chinese segmentation and new word detection using conditional random fields," Proc. Coling 2004, pp.562–568, COLING, Aug. 2004.
[11] R. Zhang, G. Kikui, and E. Sumita, "Subword-based tagging for confidence-dependent Chinese word segmentation," Proc. COLING/ACL 2006, pp.961–968, ACL, July 2006.