A Syllable-based Technique for Word Embeddings of Korean Words

Sanghyuk Choi[*] and **Taeuk Kim** and **Jinseok Seol** and **Sang-goo Lee**
Department of Computer Science and Engineering
Seoul National University
{sanghyuk, taeuk, jamie, sglee}@europa.snu.ac.kr

Abstract

Word embedding has become a fundamental component to many NLP tasks such as named entity recognition and machine translation. However, popular models that learn such embeddings are unaware of the morphology of words, so it is not directly applicable to highly agglutinative languages such as Korean. We propose a syllable-based learning model for Korean using a convolutional neural network, in which word representation is composed of trained syllable vectors. Our model successfully produces morphologically meaningful representation of Korean words compared to the original Skip-gram embeddings. The results also show that it is quite robust to the Out-of-Vocabulary problem.

1 Introduction

Continuous word representation has been a fundamental ingredient to many NLP tasks with the advent of simple and successful approaches such as Word2Vec (Mikolov et al., 2013a) and GloVe (Pennington et al., 2014). Although it has been verified that they are effective in formulating semantic and syntactic relationship between words, there are some limitations. First, they are only available to words in pre-defined vocabulary thus prone to the Out-of-Vocabulary(OOV) problem. Second, they cannot utilize subword information at all because they regard word as a basic unit. Those problems become more magnified when applying word-based methods to agglutinative languages such as Korean, Japanese, Turkish, and Finnish. In this work, we propose a new model

---

[*] Portions of this research were done while the author was a student at Seoul National University.

that utilizes syllables as basic components of word representation to alleviate the problems, especially for Korean. In our experiment, we confirm that our model constructs representation of words which contains a semantic and syntactic relationship between words. We also show that our model can handle OOV problem and capture morphological information without dedicated analysis.

2 Related Work

Recent works that utilize subword information to construct word representation could be largely divided into two families: The models that use morphemes as a component and the others taking advantage of characters.

Morpheme-based representation models

A morpheme is the smallest unit of meaning in linguistics. Therefore, there are many researches that consider morphemes when building word representations (Luong et al., 2013; Botha and Blunsom, 2014; Cotterell and Schütze, 2015).

Luong et al. (2013) applies a recursive neural network over morpheme embeddings to obtain word embeddings. Although morpheme-based models are good at capturing semantics, one major drawback is that most of them require manually annotated data or an explicit morphological analyzer which could introduce unintended errors. Our model doesn't need such a preprocessing.

Character-based representation models

Recently, utilizing information from characters has become one of the active NLP research topics. One way to extract knowledge from a sequence of characters is using character n-grams (Wieting et al., 2016; Bojanowski et al., 2016).

Bojanowski et al. (2016) suggests an approach based on the Skip-gram model (Mikolov et al., 2013a), where the model sums character n-gram vectors to represent a word. On the other hand,
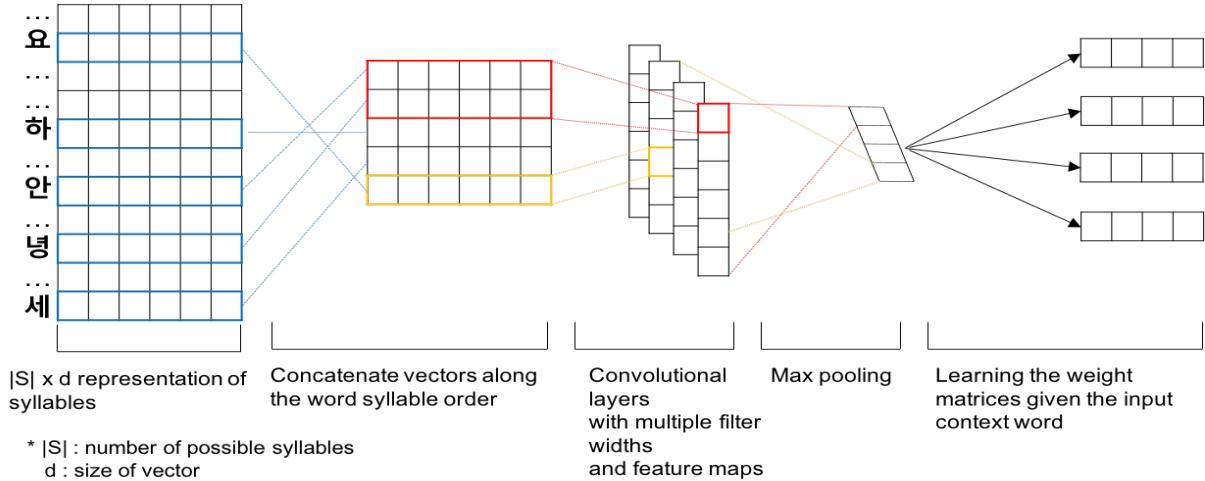
Figure 1: Overall architecture of our model. Each syllable is a d-dimensional vector. For a given word '안녕하세요' (hello, *annyeonghaseyo*), we concatenate vectors according to syllable order in word. After passing through the convolutional layer and max pooling layer, word representation is produced. All parameters are jointly trained by Skip-gram scheme.

there are some approaches (Dos Santos and Gatti, 2014; Ling et al., 2015; Santos and Guimaraes, 2015; Zhang et al., 2015; Kim et al., 2016; Jozefowicz et al., 2016; Chung et al., 2016) in which word representations are composed of character embeddings via deep neural networks such as convolutional neural networks (CNN) or recurrent neural networks (RNN).

Kim et al. (2016) introduces a language model that aggregates subword information through a character-level CNN. Models based on characters have shown competitive results on many tasks. A problem of character-based models is that characters themselves have no semantic meanings so that models often concentrate on only local syntactic features of words. To avoid the problem, we select syllables which have fine-granularity like a character but has its own meaning in Korean as a basic component of the representation of words.

3 Proposed Model

Characteristics of Korean Words

Morphologically, unlike many other languages, a Korean word (*Eojeol*) is not just a concatenation of characters. It is constructed by the following hierarchy: a sequence of syllables (*Eumjeol*) forms a word, and the composition of 2 or 3 characters (*Jaso*) forms a syllable (Kang and Kim, 1994).

In linguistics, Korean language is categorized as an agglutinative language, where each word is made of a set of morphemes. To complete the Korean word (*Eumjeol*), a root morpheme must be combined with a bound morpheme (*Josa*), or a postposition (*Eomi*). This derivation produces about 60 different forms of the similar meaning, which causes the explosion of vocabulary. For the same reason, the number of occurrences of each word is relatively small even with a large corpus, which prevents the model from an efficient learning. Thus, most of the Korean word representation models use morphemes as an embedding unit, though it requires an additional preprocessing. The problem is that errors coming from an immature morpheme analyzer might be propagated to the word representation model. Moreover, a single Korean syllable possess a semantic meaning. For example, the word '대학'(college, *daehag*) is a composition of '대'(big, or great, *dae*) and '학'(learn, or a study, *hag*). Therefore, our model regards syllables as embedding units rather than words or morphemes. For instance, the representations of '나는'(I am, *naneun*), '나의'(my, *naui*), or '나에게'(to me, *na-ege*) are constructed by leveraging the same syllable vector '나'(I, *na*).

Syllable-based Representation

Similar to (Kim et al., 2016), let $\mathcal{S}$ be a set of all Korean syllables. We embed each syllables into d-dimensional vector space, so that $Q \in \mathbb{R}^{d \times |\mathcal{S}|}$ becomes a syllable embedding ma-

trix. Let $(s_1, s_2, ..., s_l)$ denote a word $t \in V$ which consists of l syllables, t is represented by concatenating syllable vectors as a column vector: $(Qs_1, Qs_2, ..., Qs_l) \in \mathbb{R}^{d \times l}$. Then we apply a convolution filter $H \in \mathbb{R}^{d \times w}$ having a width w, we get a feature map $f^t \in \mathbb{R}^{l-w+1}$. For filters whose widths are more than 1, they need a zero padding when processing words coming from only a single syllable.

In detail, for the given filter H, the feature map can be calculated as follows:

$$f_i^t = \tanh(\langle (Qs_i, ..., Qs_{i+w-1}), H \rangle + b) \quad (1)$$

where $\langle A, B \rangle = \mathrm{tr}(AB^\mathsf{T})$ denoting Frobenius inner product. We then apply a max pooling $y^t = \max_i f_i^t$ to extract the most important feature. By using multiple filters, namely $H_1, H_2, ..., H_h$, we get a final representation $y^t = (y_1^t, ..., y_h^t)$ for the word t.

For training, we adopt Skip-gram (Mikolov et al., 2013b) method with negative sampling so that for a given center word y^t, we maximize the log-probability of predicting context word y^c. We jointly train syllable embedding matrix and convolution filters all together. Figure 1 shows overall architecture of our model.

4 Experiments and Results

Datasets and Baselines

The Experiments are performed on a randomly sampled subset of Korean News corpus collected from 2012 to 2014, containing approximately 2.7M tokens, 11k vocabulary, and 1k syllables. We compare our model to the original *skip-gram model with negative sampling* (Mikolov et al., 2013b) as a baseline.

Implementation details

For all experiments, we use the following common parameters for both our model and baseline. We use vector representations of dimension 320, the size of window is 4 and the negative-sampling parameter is 7. We train over twelve epochs. In our model, the dimension of syllable embedding is 320. Empirically, using filters with size 1~4 was enough since most of Korean words are composed of 2~4 syllables[1].

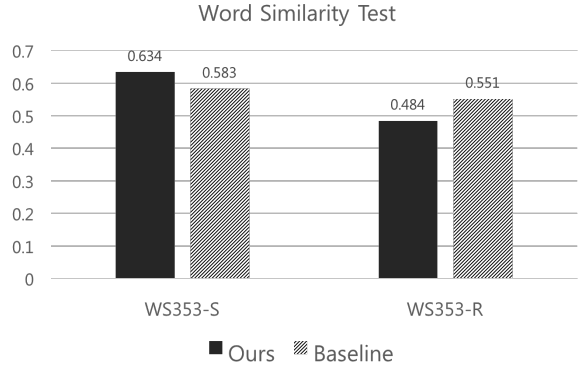[1] About 95% of words in a training set had a length less than 5.

Figure 2: Test result on translated WordSim353 dataset. It contains similarity and relatedness test and measured by Pearson correlation. Our model outperformed the baseline in similarity task.

4.1 Quantitative Evaluation

We use the WordSim353 dataset (Finkelstein et al., 2001; Agirre et al., 2009) for the word similarity and relatedness task. As WordSim353 dataset is an English data, we translated it into Korean. The quality of the word vector representation is evaluated by computing Pearson correlation coefficient between human judgment scores and the cosine similarity between word vectors.

The graph in Figure 2 shows that our model outperforms the baseline on WS353-Similarity dataset. We estimated it since a lot of similar words share the same syllable(s) in Korean. On the other hand, on WS353-Relatedness, the performance is not as good in comparison with the similarity task. We presume that leveraging syllables on computing representations can be a noise among related words without common syllables.

4.2 Qualitative Evaluation

Out-Of-Vocabulary Test

Since our model uses syllable vectors when computing word representation, it is possible to achieve representation of OOV words by combining syllables. To evaluate the representations of OOV words, we manually chose 4 newly coined words not appear in training data (Table 1). These words were derived from original words. For example, '구글신'(God Google, *gugeulsin*) is derived from '구글'(Google, *gugeul*) and '갤노 트'(Gal'Note, *gaelnoteu*) is a abbreviation form of '갤럭시노트'(Galaxy Note, *gaelleogsinoteu*). Morphologically, two of them concatenate additional syllables to the original word, and the other

38

Original word	Newly coined word
구글 (Google, *gugeul*)	구글신 (God google, *gugeulsin*)
이득 (Profit, *ideug*)	개이득 (Real profit, *gaeideug*)
퇴근 (Leave work, *toegeun*)	퇴근각 (Time to leave work, *toegeungag*)
갤럭시노트 (Galaxy Note, *gaelleogsinoteu*)	갤노트 (Gal'Note, *gaelnoteu*)

Table 1: 4 newly coined words in Korean which did not appear in training data. Proposed model successfully recognized stem from the original word, and predicted it as the most similar word.

two remove some syllables.

We examined the nearest neighbor of the representations of OOV words, and confirmed that each original word vector is placed in the nearest distance. It is no wonder since almost every newly coined word keeps the syllables of original word with their positions fixed.

Morphological Representation Test

We now evaluate our model on language morphology by observing how word representation leverages morphological characteristics. As mentioned above, the process of forming a sentence of Korean is totally different from many other languages. In case of Korean, a word can function in the sentence only if it is combined with the bound morpheme. For example, '서울을'(of Seoul, *seoul-eul*) is a combination of full morpheme '서울'(Seoul, *seoul*) + bound morpheme '을'(of, *eul*).

To compare how models learn the morphological characteristics, we randomly sampled hundred words and the same words combined with certain postposition('을', *eul*) from the training data. The graph in Figure 3 shows this result more clearly. We can observe that words forming the discriminative parallel clusters against postposition-combined-words while the baseline doesn't.

5   Conclusion

We present a syllable-based word representation model experimented with Korean, which is one of morphologically rich languages. Our model keeps
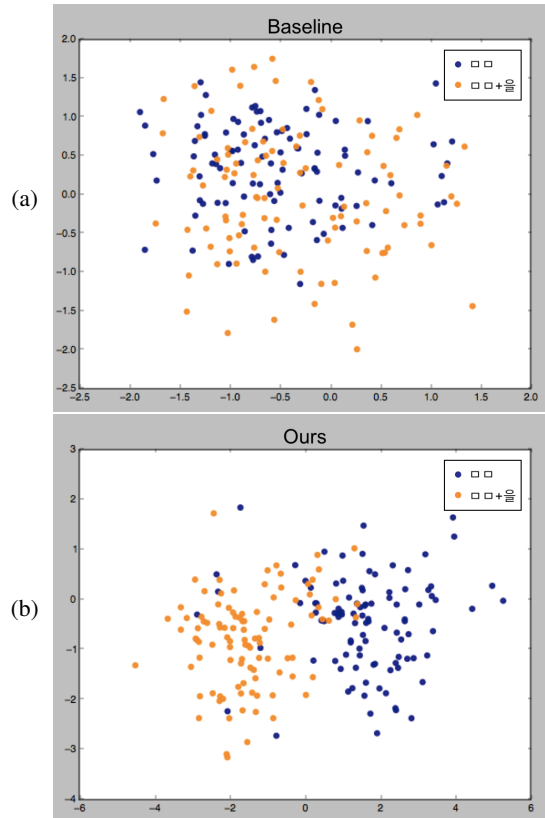

Figure 3: PCA projections of vector representation of 100 randomly sampled pairs of word. Each pair is composed of a word and the same word with postposition. In (b), our model shows that words forming the discriminative parallel clusters against postposition-combined-words.

the characteristics of Skip-gram models, in which word representation learns from context words. It also takes into account the morphological characteristics by sharing parameters between the words that contain common syllables. We demonstrate that our model is competitive on quantitative evaluations. Furthermore, we show that the model can handle OOV words, and capture morphological relationships. As a future work, we have a plan to expand our model so that it can utilize overall information extracted from words, morphemes and characters.

Acknowledgments

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 19–27.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606* .

Jan Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. In *International Conference on Machine Learning*. pages 1899–1907.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. *arXiv preprint arXiv:1603.06147* .

Ryan Cotterell and Hinrich Schütze. 2015. Morphological word-embeddings. In *HLT-NAACL*. pages 1287–1292.

Cícero Nogueira Dos Santos and Maira Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *COLING*. pages 69–78.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*. ACM, pages 406–414.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410* .

Seung-Shik Kang and Yung Taek Kim. 1994. Syllable-based model for the korean morphology. In *Proceedings of the 15th conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, pages 221–226.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Wang Ling, Tiago Luís, Luís Marujo, Ramón Fernandez Astudillo, Silvio Amir, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Finding function in form: Compositional character models for open vocabulary word representation. *arXiv preprint arXiv:1508.02096* .

Thang Luong, Richard Socher, and Christopher D Manning. 2013. Better word representations with recursive neural networks for morphology. In *CoNLL*. pages 104–113.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* .

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. pages 3111–3119.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. volume 14, pages 1532–1543.

Cicero Nogueira dos Santos and Victor Guimaraes. 2015. Boosting named entity recognition with neural character embeddings. *arXiv preprint arXiv:1505.05008* .

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2016. Charagram: Embedding words and sentences via character n-grams. *arXiv preprint arXiv:1607.02789* .

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*. pages 649–657.