Segmentation-free compositional n-gram embedding

Geewook Kim and Kazuki Fukui and Hidetoshi Shimodaira
Department of Systems Science, Graduate School of Informatics, Kyoto University
Mathematical Statistics Team, RIKEN Center for Advanced Intelligence Project
{geewook, k.fukui}@sys.i.kyoto-u.ac.jp, shimo@i.kyoto-u.ac.jp

Abstract
Applying conventional word embedding models to unsegmented languages, where word boundary is not clear, requires word segmentation as preprocessing. However, word segmentation is difficult and expensive to conduct without errors. Segmentation error degrades the quality of word embeddings, leading to performance degradation in downstream applications. In this paper, we propose a simple segmentation-free method to obtain unsupervised vector representations for words, phrases and sentences from an unsegmented raw corpus. Our model is based on subword information skip-gram model, but embedding targets and contexts are character n-grams instead of segmented words. We consider all possible character n-grams in a corpus as targets, and every target is modeled as the sum of its compositional sub-n-grams. Our method completely ignores word boundaries in a corpus and is not word-segmentation dependent. This approach may sound reckless, but it was found to work well through experiments on real-world datasets and benchmarks.

1 Introduction
Most existing word embedding methods (Bengio et al., 2003; Le and Mikolov, 2014; Pennington et al., 2014) take a sequence of words, or tokens, as their input. However, previous studies have claimed a harmful impact which comes from tokenization (Chung et al., 2016; Oshikiri, 2017). Some languages are morphologically rich with large collection of tokens. The number of unique tokens can also be increased by a vast amount of misspelling in real-world datasets. The explicit tokenization step considers all unique tokens (or character strings) independently, and that makes it infeasible to cover all unique tokens. Moreover, tokenization fails to capture the structure that involves multiple tokens. Although, there are some heuristics to detect phrases (Mikolov et al., 2013), it is still hard to perfectly cover everything.

These problems become more significant in unsegmented languages, such as Chinese and Japanese, where word boundary is not explicitly specified in text. A sequence of tokens is obtained by word segmentation tools from a raw character corpus. The accuracy of word segmentation tools strongly depends on the richness of dictionaries, and segmentation errors influence the performances or accuracies of subsequent processes (Xu et al., 2004). For example, poor dictionaries are undesirable when dealing with SNS data containing a vast amount of neologisms. However, rich dictionaries are not always available, and maintaining them up-to-date to cover neologisms is also expensive.

Another problem of word embedding with explicit tokenization step is the existence of Out-Of-Vocabulary (OOV) words. Due to tokenization error (or wrong segmentation), we may lose some words in training data. In addition, newly given real-world datasets may include a lot of unseen words and phrases. Practically, OOV words in a corpus are replaced with a special token representing OOV. The larger OOV rate in a corpus affects the accuracies of downstream tasks (Sun et al., 2005).

In recent years, an increasing number of studies have investigated character-level models with sub-words in both unsupervised (Bojanowski et al., 2017; Pagliardini et al., 2018) and supervised learning (Zhang et al., 2015; Sennrich et al., 2016; Wieting et al., 2016; Lee et al., 2017). In these models, the notion of vocabularies is extended to include sub-words. By enriching the information of the word, sub-words are useful for capturing morphological changes (Bojanowski et al., 2017) and the meaning of short phrases (Wieting et al., 2016). In addition, OOV (or unseen) words can
be composed from sub-words, which are present at training (Bojanowski et al., 2017).

In this paper, we propose a simple unsupervised method of character \( n \)-gram embedding for unsegmented languages, where the segmentation step is completely omitted thus words, phrases and sentences are treated seamlessly. Our model considers all possible character \( n \)-grams as embedding targets in a corpus. Each \( n \)-gram is explicitly modeled as a composition of its sub-\( n \)-grams just like each word is modeled as a composition of sub-words in the subword information skip-gram model (SISG) (Bojanowski et al., 2017). Our segmentation-free compositional \( n \)-gram embedding is referred to as SCNE in this paper.

This kind of approach that does not consider any word boundaries for unsegmented languages may sound reckless since the embedding targets can include a lot of wrong boundaries. However, we found that we can compose vector representations for words and sentences with good quality by summing up the representations of their substrings. Oshikiri (2017) also proposed a segmentation-free word embedding for unsegmented languages, which is called as Sembei, but the word and sentence vectors of our model are enriched by their substrings, and hence our method may achieve better performance in downstream tasks.

2 Model

Our method SCNE is inspired by recent successful sub-word models (Bojanowski et al., 2017; Zhang et al., 2015) as well as by the segmentation-free character \( n \)-gram embedding for unsegmented languages (Oshikiri, 2017).

Vector representation of target \( n \)-gram is defined as follows. Let \( x_1x_2 \cdots x_N \) be a raw unsegmented corpus of \( N \) characters. For a range of index \( t = (i, j), 1 \leq i \leq j \leq N \), the substring \( x_ix_{i+1} \cdots x_j \) is denoted as \( x_t \). We first count occurrences of \( n \)-gram \( s = s_1s_2 \cdots s_n \) in the raw corpus as \( x_t = s \) with length \( n = j-i+1 \leq n_{\text{max}} \). Then \( n \)-gram vocabulary \( V = \{ s \} \) is constructed by collecting \( M \)-most frequent \( n \)-grams with \( n \leq n_{\text{max}} \). For any target \( n \)-gram \( w = w_1w_2 \cdots w_n \), \( n \geq 1 \), the sequence of its sub-\( n \)-grams included in the vocabulary is \( S(w) = \{ w_{(i,j)} \in V \mid 1 \leq i \leq j \leq n \} \), and the vector representation \( v_w \in \mathbb{R}^d \) is computed as

\[
v_w = \sum_{s \in S(w)} z_s,
\]

\( z_s \in \mathbb{R}^d \) is the free word embedding for unsegmented languages, where the segmentation step is completely omitted thus words, phrases and sentences are treated seamlessly. For any target \( n \)-gram \( w = w_1w_2 \cdots w_n \), \( n \geq 1 \), the sequence of its sub-\( n \)-grams included in the vocabulary is \( S(w) = \{ w_{(i,j)} \in V \mid 1 \leq i \leq j \leq n \} \), and the vector representation \( v_w \in \mathbb{R}^d \) is computed as

\[
\sum_{t \in D} \left( \sum_{c \in C(t)} \log \sigma (v_{x_t}^\top u_x) + \sum_{s \sim P_{\text{neg}}} \log \sigma (-v_{x_t}^\top u_s) \right),
\]

where \( \sigma \) is the sigmoid function, and \( D = \{ (i, j) \mid 1 \leq i \leq j \leq N, j - i + 1 \leq n_{\text{target}} \} \) is the set of target ranges considered in the objective. The set of context ranges is defined as \( C((i, j)) = \{ (i', j') \mid x_{(i', j')} \in V, j' = i - 1 \text{ or } j' = j + 1 \} \). Note that we do not employ compositional form for the contexts. The negative sampling distribution \( P_{\text{neg}} \) of \( s \in V \) is proportional to the frequency in the corpus. Model parameters \( z_s, u_s \in \mathbb{R}^d \), \( s \in V \), are learned by maximizing the objective. We set \( n_{\text{target}} = n_{\text{max}} \) in the experiments. Large \( n_{\text{max}} \), such as 16, contributes to capturing sentence-level representations while 4 to 8 is enough for word embeddings.

3 Experiments

We use a standard set of benchmark tasks to evaluate our trained models at both word and sentence level on three different languages: Chinese,
Japanese, and Korean\footnote{Although Korean has spacing, word boundaries are not obviously determined by space.}. Through the following two word-level tasks and one sentence-level task, we investigate the qualities of vector representations and their usefulness in practical applications. We will make the C++ implementation of our method and pre-trained models available open-source.

### 3.1 Baseline systems

As baseline systems, we use C-BOW, Skip-gram (Mikolov et al., 2013), Subword Information Skip-gram (SISG) (Bojanowski et al., 2017) and Segmentation-free word embedding for unsegmented languages (Sembei) (Oshikiri, 2017) for the word-level tasks. For the sentence-level task, baselines are PV-DBOW, PV-DM (Le and Mikolov, 2014) and Sent2vec (Pagliardini et al., 2018). In addition, we also test sentence embedding baselines obtained by simple averaging of word embeddings over the sentence, denoted as C-BOW\*, Skip-gram\* and SISG\*. We also test a variant of Sembei, denoted by Sembei\*, which calculates word or sentence embeddings by simple averaging of sub-\textit{n}-gram embeddings, to see whether our model provides more effective compositional \textit{n}-gram embeddings compared to the previously proposed non-compositional model.

### 3.2 Evaluation tasks

**Word similarity task:** Our model and the baselines are trained on portions of Wikipedia of increasing size to see the effect of the size of the training data. For pairs of words, the cosine similarity between embeddings is measured by human judgment, and the quality is measured by Spearman rank correlation. Most of the settings are the same as that of Bojanowski et al. (2017). Two widely-used benchmark datasets are used: Chinese word similarity dataset (Jin and Wu, 2012), which contains 297 pairs of words, and Japanese word similarity dataset (Sakaizawa and Komachi, 2017), which contains 4429 pairs of words.

Conventional word embedding methods, C-BOW, Skip-gram, and Sembei, cannot provide the embeddings of OOV words in the test data. In contrast, SISG and our model can compute representations for almost all words, since both methods learn compositional \textit{n}-gram features. In order to show comparable results, we use the null vector for these OOV words following Bojanowski et al. (2017).

**Noun category prediction task:** We use 100MB of SNS data, Sina Weibo for Chinese and Twitter for Japanese and Korean, as training corpora. For evaluating the learned embeddings, noun words, including neologisms, and their categories are extracted from Wikidata with the predetermined semantic category set\footnote{Two food, song, music band name, manga, fictional character name, television series, drama, chemical compound, disease, taxon, city, island, country, year, business enterprise, public company, profession, university, language, book}. For each category, a logistic regression classifier is trained from the vector representations, where unseen words are skipped in training and treated as errors in testing.

**Sentiment analysis task:** Movie review datasets are used for evaluating sentence embeddings. We use 101,114, 55,837 and 200,000 movie reviews and their rating scores from Yahoo\footnote{https://github.com/fychao/ChineseMovieReviews}, Yahoo\footnote{https://github.com/dennybritz/sentiment-analysis/tree/master/data} and Naver Movies\footnote{https://github.com/e9t/nsmc} for Chinese, Japanese and Korean, respectively. Each review is labeled as positive or negative by its rating score. We simply concatenate all reviews into a single document and use it as the training corpus. At the testing phase, 5,000 positive and 5,000 negative reviews are randomly selected and used for evaluation. In this experiment, we only include unsupervised sentence embedding models as baselines, i.e. not task-specific model, to ensure coherence.

### 3.3 Common settings

The dimension of word vector is \( d = 200 \) and the number of negative samples is \( k = 10 \) for all methods. The number of epochs is 10 for Sembei and SCNE, and 20 for the other baselines. The \textit{n}-gram vocabulary size \( M = 2 \times 10^6 \) is used for SISG, Sembei and SCNE. The initial learning rate \( \gamma = 10^{-2} \) is used for Sembei and SCNE. \( n_{\text{max}} = 8 \) is used for Sembei and SCNE in the word-level tasks, while \( n_{\text{max}} = 16 \) is also used in the sentence-level task. The other hyper-parameters of the baselines, such as window size, are adjusted via grid search.

For the conventional word-segmentation dependent baselines, we employ widely-used word segmentation tools with only basic dictionary (\textit{basic}) or using rich dictionary together (\textit{rich}), to see the...
Table 1: Noun category prediction accuracies (higher is better) and coverages [%] (in parentheses, higher is better).

|                      | Compositional | Seg.-free | Chinese (All) | Japanese (All) | Korean (All) |
|----------------------|---------------|-----------|---------------|----------------|--------------|
|                      |               |           |               | Intersec.      | Intersec.    |
| C-BOW<sub>basic</sub> | 4.5 (9)       | 51.8      | 4.9 (12)      | 40.5           | 3.8 (11)     |
| C-BOW<sub>rich</sub>  | 4.6 (9)       | 51.1      | 7.5 (20)      | 39.9           | 3.9 (12)     |
| Skip-gram<sub>basic</sub> | 4.8 (9)    | 53.8      | 5.5 (12)      | 39.2           | 3.9 (11)     |
| Skip-gram<sub>rich</sub> | 5.0 (9)     | 54.2      | 7.7 (20)      | 38.1           | 3.9 (12)     |
| SISG<sub>basic</sub>  | ✓             | 71.1 (100)| 66.4 (100)    | 48.7           | 62.9 (100)   |
| SISG<sub>rich</sub>   | ✓             | 70.6 (100)| 67.2 (100)    | **49.1**       | 62.9 (100)   |
| Sembei<sup>*</sup><sub>n<sub>max</sub>=8</sub> | ✓             | 4.5 (7)   | 4.9 (10)      | 42.6           | 5.0 (13)     |
| SCNE<sub>n<sub>max</sub>=8</sub> | ✓             | 72.3 (100)| 66.3 (100)    | 42.8           | 64.2 (100)   |

Effect of rich resources.

In the downstream supervised tasks, vector representations are combined with a logistic regression classifier. We repeat training and testing of the classifier for 10 times, while the dataset is randomly split into train (60%) and test (40%) sets at each time. We adopt mean accuracy as the evaluation metric.

4 Results

Word similarity task: The results are shown in Figure 1. The first 10, 50, 100, 200, 300MB of Wikipedia corpus is used in each language setting. The number of OOV words in the benchmarks grow as the dataset shrinks, and therefore the performances of C-BOW, Skip-gram and Sembei are necessarily degraded. We can see that, for all datasets and all corpus sizes, our proposed method SCNE outperforms the baselines. We can see that the proposed approach provides high-quality word vectors even when using very small training dataset, which is crucial for practical real-world settings where rich data is not available.

Noun category prediction task: The results are reported in Table 1. Since the set of covered nouns (non-OOV words) depends on the methods, we calculate accuracies in two ways for a fair comparison: Using all the nouns and using the intersection of the covered nouns. We notice that our proposed model SCNE achieved the highest prediction accuracies in most of the settings. These results demonstrate the efficacy of the proposed method both quantitatively and qualitatively.

Sentiment analysis task: The results are reported in Table 2. The classification accuracies show that our method SCNE is very effective in the sentence-level application. Furthermore, the larger n<sub>max</sub> contributes to the performance improvement by allowing our model to capture composed representations for longer phrases or sentences, whereas there is little improvement in Sembei<sup>*</sup> with larger n<sub>max</sub>. This fact shows the efficacy of compositional n-gram features in our model.

Table 2: Sentiment classification accuracies.

|          | Seg.-free | Chinese | Japanese | Korean |
|----------|-----------|---------|----------|--------|
| C-BOW<sub>basic</sub> | 84.91     | 86.88   | 83.41    |
| C-BOW<sub>rich</sub>  | 85.38     | 87.21   | 83.73    |
| Skip-gram<sub>basic</sub> | 85.74    | 86.84   | 84.77    |
| Skip-gram<sub>rich</sub> | 86.11    | 87.11   | 84.77    |
| SISG<sub>basic</sub>  | 85.59     | 86.90   | 83.87    |
| SISG<sub>rich</sub>   | 85.78     | 87.20   | 84.19    |
| PV-DBOW<sub>basic</sub> | 83.85    | 84.39   | 60.97    |
| PV-DBOW<sub>rich</sub> | 83.61    | 84.75   | 58.99    |
| PV-DM<sub>basic</sub> | 76.88     | 81.25   | 60.98    |
| PV-DM<sub>rich</sub>  | 77.60     | 82.18   | 58.70    |
| Sent2vec<sub>basic</sub> | 84.90    | 87.06   | 82.49    |
| Sent2vec<sub>rich</sub> | 85.61    | 87.29   | 82.59    |
| Sembei<sup>*</sup><sub>n<sub>max</sub>=8</sub> | ✓         | 83.19   | 80.80    | 74.98   |
| Sembei<sup>*</sup><sub>n<sub>max</sub>=16</sub> | ✓         | 83.41   | 80.45    | 74.09   |
| SCNE<sub>n<sub>max</sub>=8</sub> | ✓         | 87.07   | 86.96    | 84.25   |
| SCNE<sub>n<sub>max</sub>=16</sub> | ✓         | **87.94** | **88.32** | **86.26** |

5 Conclusion

We proposed segmentation-free compositional n-gram embedding (SCNE), a new unsupervised method to acquire general-purpose vector representations of words, phrases and sentences, whereas there is little improvement in Sembei<sup>*</sup> with larger n<sub>max</sub>. This fact shows the efficacy of compositional n-gram features in our model.
References

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. Journal of machine learning research, 3(Feb):1137–1155.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5:135–146.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1693–1703. Association for Computational Linguistics.

Peng Jin and Yunfang Wu. 2012. Semeval-2012 task 4: Evaluating chinese word similarity. In *SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012), pages 374–377. Association for Computational Linguistics.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32, ICML’14, pages II–1188–II–1196. JMLR.org.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully character-level neural machine translation without explicit segmentation. Transactions of the Association for Computational Linguistics, 5:365–378.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS’13, pages 3111–3119, USA. Curran Associates Inc.

Takamasa Oshikiri. 2017. Segmentation-free word embedding for unsegmented languages. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 767–772. Association for Computational Linguistics.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. In NAACL 2018 - Conference of the North American Chapter of the Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543. Association for Computational Linguistics.

Yuya Sakaizawa and Mamoru Komachi. 2017. Construction of a japanese word similarity dataset. CoRR, abs/1703.05916.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1715–1725. Association for Computational Linguistics.

Chengjie Sun, Chang-Ning Huang, Xiaolong Wang, and Mu Li. 2005. Detecting segmentation errors in chinese annotated corpus. In Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2016. Charagram: Embedding words and sentences via character n-grams. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1504–1515. Association for Computational Linguistics.

Jia Xu, Richard Zens, and Hermann Ney. 2004. Do we need chinese word segmentation for statistical machine translation? In ACL SIGHAN Workshop 2004, pages 122–128, Barcelona, Spain. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, NIPS’15, pages 649–657, Cambridge, MA, USA. MIT Press.