Segmentation-Free Word Embedding for Unsegmented Languages*

Takamasa Oshikiri
Graduate School of Engineering Science, Osaka University
RIKEN Center for Advanced Intelligence Project
CyberAgent, Inc.
mail@oshikiri.org

Abstract

In this paper, we propose a new pipeline of word embedding for unsegmented languages, called *segmentation-free word embedding*, which does not require word segmentation as a preprocessing step. Unlike space-delimited languages, unsegmented languages, such as Chinese and Japanese, require word segmentation as a preprocessing step. However, word segmentation, that often requires manually annotated resources, is difficult and expensive, and unavoidable errors in word segmentation affect downstream tasks. To avoid these problems in learning word vectors of unsegmented languages, we consider word co-occurrence statistics over all possible candidates of segmentations based on frequent character n-grams instead of segmented sentences provided by conventional word segmenters. Our experiments of noun category prediction tasks on raw Twitter, Weibo, and Wikipedia corpora show that the proposed method outperforms the conventional approaches that require word segmenters.

1 Introduction

Word embedding, which learns dense vector representation of words from large text corpora, has received much attention in the natural language processing (NLP) community in recent years. It is reported that the representation of words well captures semantic and syntactic properties of words (Bengio et al., 2003; Mikolov et al., 2013; Pennington et al., 2014), and is useful for many downstream NLP tasks, including part-of-speech tagging, syntactic parsing, and machine translation (Huang et al., 2011; Socher et al., 2013; Sutskever et al., 2014).

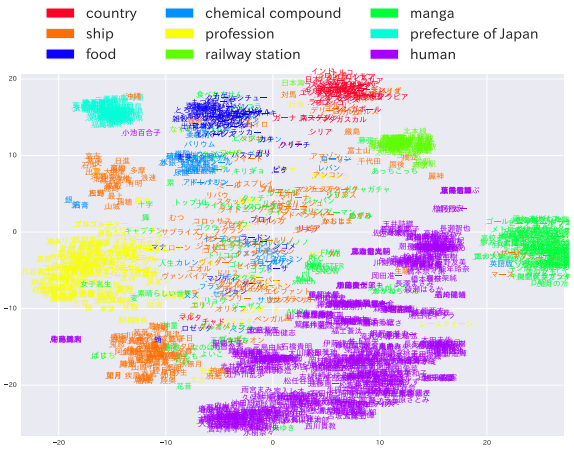

Figure 1: t-SNE projections of vector representation of Japanese nouns that generated by our proposed method without word dictionary. These proper nouns are color-coded according to its categories which extracted from Wikidata.

In order to train word embedding models on a raw text corpus, we have to do word segmentation as a preprocessing step. In space-delimited languages such as English and Spanish, simple rule-based and co-occurrence-based approaches offer reasonable segmentations. On the other hands, these approaches are impractical for unsegmented languages such as Chinese, Japanese, and Thai. Therefore, machine learning-based approaches are widely used in NLP for unsegmented languages. Conditional random field (CRF)-based supervised word segmentation (Kudo et al., 2004; Tseng et al., 2005) is still the most used one in Japanese and Chinese NLP (Prettenhofer and Stein, 2010; Funaki and Nakayama, 2015; Ishiwatari et al.,

* This work was done while the author was at Shimodaira laboratory, Division of Mathematical Science, Graduate School of Engineering Science, Osaka University, and Mathematical Statistics Team, RIKEN Center for Advanced Intelligence Project.

2015; Nakazawa et al., 2016).

However, there are some problems for these supervised word segmentation as a preprocessing step of a word embedding pipeline. First, they require language-specific manually annotated resources such as word dictionaries and segmented corpora. Since these manually annotated resources are typically unavailable for domain-specific corpora (e.g. Twitter or Weibo corpora that contain many neologisms and informal words), we have to create manually annotated resources if we need. Second, they cannot take advantage of word occurrence frequencies in a corpus. Even though a certain proper noun (e.g. "老人と海" (*The Old Man and the Sea*)) occurs frequently in a corpus, word segmenters will continue to split the proper noun erroneously (e.g. "老人/と/海" (a old man / and / a sea)) if it is not registered in the word dictionary. Because of segmentation errors incurred by these problems, the downstream word embedding model cannot learn vector representation of proper nouns, neologisms, and informal words.

In this paper, in order to learn word vectors from a raw text corpus while avoiding the above problems, we propose a new word segmentation-free pipeline for word embedding, referred to as *segmentation-free word embedding (sembei)*. Our framework first enumerates all possible segmentations (referred to as a *frequent n-gram lattice*) based on character n-grams that frequently occurred in the raw corpus, and then learns n-gram vectors from co-occurrence frequencies over the frequent n-gram lattice. Using the general idea of segmentation-free word embedding, we can extend existing word embedding models. Specifically, in this paper, we propose a segmentation-free version of the widely used skip-gram model with negative sampling (SGNS) (Mikolov et al., 2013), which we refer to as *SGNS-sembei*.

Although the frequent character n-grams necessarily include many non-words (i.e. n-grams that are not words), remarkably, our results show that nearest neighbor search works well for frequent words and even proper nouns (e.g. nearest neighbors of n-gram "ドイツ" (Germany) are "中国" (China), "イギリス" (United Kingdom), etc.). This observation suggests that we can use the proposed method for automatic acquisition of synonyms from large raw text corpora.

We conduct experiments on a noun category prediction task on several corpora and observe that our method outperforms the conventional approaches that use word segmenters. Fig. 1 shows a t-SNE projection of vector representation of Japanese nouns which is learned from only a raw Twitter corpus. We can see that the proposed method can learn vector representation of these nouns, and the learnt representation achieves good separation based on their categories.

2 Related Work

There are some representation models that do not rely on any segmenters. Dhingra et al. (2016) proposed a character-based RNN model for vector representation of tweets, and Schütze (2017) proposed a new text embedding method that learns n-gram vectors from the corpus that segmented randomly and then constructs text embeddings by summing up the n-gram vectors. In the field of representation learning for biological sequences (e.g. DNA and RNA), Asgari and Mofrad (2015) applied the skip-gram model (Mikolov et al., 2013) to fixed length fragments of biological sequences. These methods mainly aim at learning vector representation of texts or biological sequences instead of words or fragments of sequences. On the other hand, in this paper, we focus on learning vector representation of words from a raw corpus of unsegmented languages.

3 Conventional Approaches to Word Embeddings

Word embedding is also commonly used in NLP for unsegmented languages (Prettenhofer and Stein, 2010; Funaki and Nakayama, 2015; Ishiwatari et al., 2015). In these studies, they usually segment a raw corpus into words using a word segmenter or a morphological analyzer, and then feed the segmented corpus to word embedding models (e.g. the skip-gram model (Mikolov et al., 2013) or the GloVe (Pennington et al., 2014)) as in the case of space-delimited languages. The flowchart of the above process is shown in the left part of Fig. 2.

3.1 The original SGNS

The original skip-gram model with negative sampling (Mikolov et al., 2013) (we refer to it as *the original SGNS*) learns vector representation of words $\boldsymbol{v}_w$ and their contexts $\tilde{\boldsymbol{v}}_c$ that minimize the

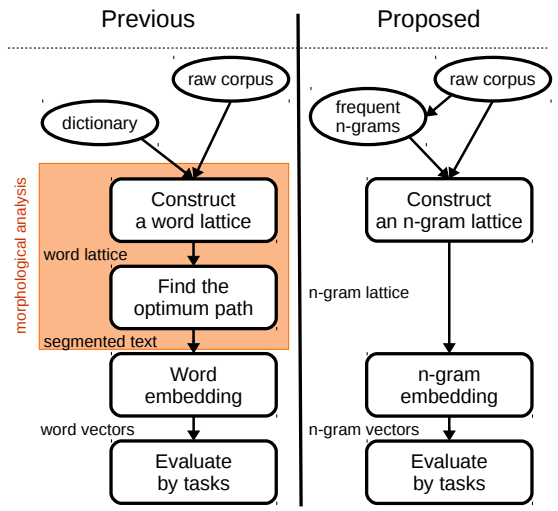

Figure 2: Flowcharts of previous and proposed pipelines. Morphological analyzers are in charge of the shaded part. Our main idea is to replace a word dictionary with a set of frequent character n-grams, and omit the indentification of the optimum path.

following objective function:

$$\underset{\{\boldsymbol{v}_w\}\cup\{\tilde{\boldsymbol{v}}_c\}}{\text{maximize}} \sum_{(w,c)\in\mathcal{D}} \log\sigma(\boldsymbol{v}_w^\top\tilde{\boldsymbol{v}}_c) + \sum_{(w,c)\in\mathcal{D}'} \log\sigma(-\boldsymbol{v}_w^\top\tilde{\boldsymbol{v}}_c) \tag{1}$$

where $\sigma(x) := (1+e^{-x})^{-1}$, $\mathcal{D}$ is a multiset (bag) of positive samples (i.e. co-occurred pairs in the corpus), and $\mathcal{D}'$ is a multiset of negative samples. This objective function is maximized using stochastic gradient descent (SGD).

4 Segmentation-Free Word Embeddings

In this section, we first introduce the general idea of *segmentation-free word embeddings* (*sembei*), and then propose a segmentation-free version of the SGNS.

While conventional word embedding approaches learn word vectors from segmented corpora that provided by word segmenters, our approach learns n-gram vectors from raw corpora, as in the right part of Fig. 2. In order to learn n-gram vectors from a raw corpus of unsegmented languages, we first construct a *frequent n-gram lattice*, which represents all possible segmentations based on frequent character n-grams of the corpus, in the same way as the construction of word lattices used in morphological analysis. Then, we learn n-gram vectors using co-occurrence statistics over the frequent n-gram lattice instead of segmented corpora as in conventional approaches.

4.1 Segmentation-Free Version of the SGNS

Here, we introduce a segmentation-free version of SGNS, referred to as *SGNS-sembei*, as an application of the idea of segmentation-free word embedding. Our method simply optimizes the original SGNS's objective function (1) with the slight modification: changing the definition of the multiset of positive samples $\mathcal{D}$.

In SGNS-sembei, $\mathcal{D}$ is redefined as the multiset of character n-gram pairs (w, c) where w and c occur adjacently in the corpus (i.e. w and c are connected in the frequent n-gram lattice). In addition, to discriminate co-occurrence with different order in the frequent n-gram lattice, we define contextual words with their relative positions to the center word as the same way as Ling et al. (2015) did.

We also redefine the multiset of negative samples $\mathcal{D}'$ using $\mathcal{D}$ in the same way as the original SGNS, and then optimize the objective function (1) using SGD.

Table 1: Examples of labels of entities (in Japanese, and in English for reference) and its categories extracted from Wikidata.

label (ja)	label (en)	category
ドイツ	Germany	country
二酸化炭素	carbon dioxide	chemical compound
消防士	firefighter	profession
アップルパイ	apple pie	food
長友佑都	Yuto Nagatomo	human

5 Experiment

In this section, we evaluate our method by the noun category prediction task on Twitter, Weibo, and Wikipedia corpora.

The C++ implementation of the proposed method is available on GitHub[1].

5.1 Settings

We used four raw text corpora: Wikipedia (Japanese), Wikipedia (Chinese), Twitter (Japanese), and Weibo (Chinese). The Wikipedia corpora consist of only a part of the Wikipedia

[1] https://github.com/oshikiri/w2v-sembei

Table 2: Micro F-scores (higher is better) and coverages [%] (in parentheses, higher is better).

	dictionary		Japanese		Chinese	
	default	Wikidata	Wikipedia	Twitter	Wikipedia	Weibo
SGNS	✓		0.896 (34)	0.761 (46)	0.889 (86)	0.766 (88)
SGNS	✓	✓	0.945 (98)	0.867 (96)	**0.891** (94)	0.765 (93)
SGNS-sembei			**0.949** (**100**)	**0.870** (**100**)	**0.891** (**100**)	**0.811** (**100**)

dumps[2] (dated on February 20th, 2017), whose HTML tags are removed. The Weiboscope corpus (Chau et al., 2013) consists of 226,841,122 posts mainly in Chinese, and we use only a part of it. The Twitter corpus consists of 17,316,968 Japanese tweets that were collected from October 26th, 2016 until November 22nd, 2016 via the Twitter Streaming API. We removed hashtags, users' id, and URL from Twitter and Weibo corpora. We extracted about 1,460k frequent n-grams[3] as the *frequent character n-grams* for our proposed method.

We extracted the noun-category pairs from the Wikidata (Vrandečić and Krötzsch, 2014) (We used the dump dated January 9th, 2017) as follows. We first extracted Wikidata entities whose headwords are also in the 1,460k frequent n-grams, and then extracted the Wikidata entities whose "`instance of`" properties are any of the predetermined category set[4], and then collected names and their categories of the entities. Examples of the extracted noun-category pairs are shown in Table 1.

We randomly split the noun-category pairs into a train (60%) and a test (40%) set. We trained linear C-SVM classifiers (Hastie et al., 2009) with the train set to predict categories from vector representation of the nouns. We performed a grid search over $(C, \text{classifier}) \in \{0.5, 1, 5, 10, 50, 100\} \times \{\text{one-vs-one}, \text{one-vs-rest}\}$ of linear SVM using the train set for each vector representation, and reported the best scores on the test set.

5.2 Baseline Systems

We compared SGNS-sembei with the conventional approaches that use the original SGNS and word segmenters. To segment the raw corpora, we used the MeCab (Kudo et al., 2004) for Japanese corpora and the Stanford Word Segmenter (Tseng et al., 2005) for Chinese corpora with their default dictionaries[5]. And we ignored the words that occur less than 5 times. We also ran these baseline systems in an ideal setting: running the word segmenters with the default dictionaries and additional dictionaries that consist of the nouns extracted in § 5.1.

We performed a grid search over $(h, t, n_{\text{neg}}) \in \{5, 8, 10\} \times \{10^{-5}, 10^{-4}, 10^{-3}\} \times \{3, 10, 25\}$ where h is the size of context window, t is the sampling threshold, and n_{neg} is the number of negative samples.

5.3 Results

In both the original SGNS and SGNS-sembei, we fixed the dimensionality of vector representation to 200 and the number of iterations to 5 in both baseline and our method. In SGNS-sembei, we used the number of negative samples $n_{\text{neg}} = 10$, size of context window $h = 1$, initial learning rate $\alpha_{\text{init}} = 0.01$.

The resulting micro F-scores and the coverages (i.e. the percentages of the noun-category pairs whose nouns' vector representation exists) are shown in Table 2, and the t-SNE (Maaten and Hinton, 2008) projections of Japanese nouns vectors learned from the Twitter corpus are shown in Fig. 1. We observed that our proposed method outperforms the conventional approaches that use word segmenters. Furthermore, the coverages of our method were higher than those of the SGNS with the default dictionary (especially in Japanese) and competitive to those of the SGNS with the default dictionary and Wikidata (which is an ideal

[2] We used {ja,zh}wiki-20170220-pages-articles1.xml in `https://dumps.wikimedia.org`

[3] In this experiment, we defined the frequent n-grams as the union of the top-k_n frequent n-grams, where n and k_n are the pre-specified numbers. And we used $n = 8$, $(k_1, \ldots, k_8) = (10000, 300000, 300000, 300000, 200000, 200000, 100000, 50000)$ for Japanese corpora, and $n = 7$, $(k_1, \ldots, k_7) = (10000, 400000, 400000, 300000, 200000, 100000, 50000)$ for Chinese corpora

[4] {country, profession, ship, railway station, food, chemical compound, prefecture of Japan, manga, human } for Japanese, and {country, profession, television series, business enterprise, city, chemical compound, taxon, human} for Chinese

[5] We use `mecab-ipadic v2.7.0` for the MeCab and `dict-chris6.ser.gz` for the Stanford Word Segmenter.

setting) even though our method does not require any manually annotated resources. We can also see that the learnt representation achieves good separation based on their categories as in Fig. 1. Nearest neighbor search using Twitter and Weibo corpora was also performed as preliminary experiments, and surprisingly, it worked well for frequent words as in Table. 3.

Table 3: Results of nearest neighbor search for frequent words

Language	Query	3-Nearest Neighbors
Japanese	ドイツ (Germany)	中国 (China), イギリス (UK), ポーランド (Poland)
	酸素 (oxygen)	水素 (hydrogen), 鉄分 (iron), 二酸化炭素 (carbon dioxide)
Chinese	德国 (Germany)	美国 (USA), 英国 (UK), 法国 (France)
	羽毛球 (badminton)	台球 (billiards), 网球 (tennis), 乒乓球 (pingpong)

6 Conclusion

We proposed segmentation-free word embedding for unsegmented languages. Although our method does not rely on any manually annotated resources, experimental results of the noun category prediction task on several corpora showed that our method outperforms conventional approaches that rely on manually annotated resources.

As an anonymous reviewer suggested, a possible direction of future work is to leverage another word segmentation approach which uses linguistic features, such as the Stanford Word Segmenter (Tseng et al., 2005) with k-best segmentations.

Acknowledgments

I would like to thank Hidetoshi Shimodaira, Thong Pham, Kazuki Fukui, and anonymous reviewers for their helpful suggestions. This work was partially supported by grants from Japan Society for the Promotion of Science KAKENHI (JP16H02789) to HS.

References

Ehsaneddin Asgari and Mohammad R. K. Mofrad. 2015. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLOS ONE*, 10(11):1–15.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Michael Chau, Chung hong Chan, and King wa Fu. 2013. Assessing censorship on microblogs in china: Discriminatory keyword analysis and the real-name registration policy. *IEEE Internet Computing*, 17:42–50.

Bhuwan Dhingra, Zhong Zhou, Dylan Fitzpatrick, Michael Muehl, and William Cohen. 2016. Tweet2vec: Character-based distributed representations for social media. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 269–274, Berlin, Germany. Association for Computational Linguistics.

Ruka Funaki and Hideki Nakayama. 2015. Image-mediated learning for zero-shot cross-lingual document retrieval. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 585–590, Lisbon, Portugal. Association for Computational Linguistics.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning*, 2 edition. Springer New York.

Fei Huang, Alexander Yates, Arun Ahuja, and Doug Downey. 2011. Language models as representations for weakly supervised nlp tasks. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 125–134, Portland, Oregon, USA. Association for Computational Linguistics.

Shonosuke Ishiwatari, Nobuhiro Kaji, Naoki Yoshinaga, Masashi Toyoda, and Masaru Kitsuregawa. 2015. Accurate cross-lingual projection between count-based word vectors by exploiting translatable context pairs. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 300–304, Beijing, China. Association for Computational Linguistics.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 230–237, Barcelona, Spain. Association for Computational Linguistics. http://taku910.github.io/mecab/.

Wang Ling, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1299–1304, Denver, Colorado. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Toshiaki Nakazawa, Chenchen Ding, Hideya Mino, Isao Goto, Graham Neubig, and Sadao Kurohashi. 2016. Overview of the 3rd workshop on asian translation. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 1–46, Osaka, Japan. The COLING 2016 Organizing Committee.

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*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Peter Prettenhofer and Benno Stein. 2010. Cross-language text classification using structural correspondence learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1118–1127, Uppsala, Sweden. Association for Computational Linguistics.

Hinrich Schütze. 2017. Nonsymbolic text representation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 785–796, Valencia, Spain. Association for Computational Linguistics.

Richard Socher, John Bauer, Christopher D. Manning, and Ng Andrew Y. 2013. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 455–465, Sofia, Bulgaria. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. 2005. A Conditional Random Field Word Segmenter for SIGHAN bakeoff 2005. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, volume 171. Jeju Island, Korea. `http://nlp.stanford.edu/software/segmenter.shtml`.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.