Improve Chinese Word Embeddings by Exploiting Internal Structure

Jian Xu, Jiawei Liu, Liangang Zhang, Zhengyu Li, Huanhuan Chen*
Department of Computer Science, University of Science and Technology of China, China
{jianxu1, ustcljw, liangang, lzy0503}@mail.ustc.edu.cn
hchen@ustc.edu.cn

Abstract

Recently, researchers have demonstrated that both Chinese word and its component characters provide rich semantic information when learning Chinese word embeddings. However, they ignored the semantic similarity across component characters in a word. In this paper, we learn the semantic contribution of characters to a word by exploiting the similarity between a word and its component characters with the semantic knowledge obtained from other languages. We propose a similarity-based method to learn Chinese word and character embeddings jointly. This method is also capable of disambiguating Chinese characters and distinguishing non-compositional Chinese words. Experiments on word similarity and text classification demonstrate the effectiveness of our method.

1 Introduction

Distributed representations of knowledge has received wide attention in recent years. Researchers have proposed various models to learn it at different granularity levels. Distributed word representations, also known as word embeddings, were learned in (Rumelhart et al., 1988; Bengio et al., 2006; Mnih and Hinton, 2009; Mikolov et al., 2013a). Larger granularity levels than words have also been investigated, including phrase level (Socher et al., 2010; Zhang et al., 2014; Yu and Dredze, 2015), sentence level (Le and Mikolov, 2014; Socher et al., 2013; Kalchbrenner et al., 2014; Kiros et al., 2015), and document level (Le and Mikolov, 2014; Hermann and Blunsom, 2014; Srivastava et al., 2013).

For language like Chinese, some smaller units than word also provide rich semantic information. For example, Chinese characters in word, Chinese radicals in character. These internal structures have been proved to be useful for Chinese word and character embeddings (Chen et al., 2015; Li et al., 2015). Chen et al. (2015) took Chinese characters in a word into account when modeling the semantic meaning of the word. They proposed a character-enhanced word embeddings model (CWE) by adding the embedding of component characters in a word with the same weight to the word embedding. However, the internal characters in a Chinese word have different semantic contributions to its meaning. Take Chinese word "青蛙" (frog) as an example. The character "青" (blue or green) is to decorate character "蛙" (frog). It is obvious that the latter character contributes more than the former one to the word meaning. In Li et al. (2015), they proposed a component-enhanced Chinese character embeddings model based on the feature that most Chinese characters are phono-semantic compounds. They considered characters and bi-characters as the basic embedding units. However, some bi-characters are meaningless, and may not form a Chinese word. These bi-characters may undermine embeddings of others.

This paper, motivated by Chen et al. (2015), exploits the internal structures of Chinese word, namely the Chinese characters. We propose a method to calculate the semantic contribution of characters to a word in a cross-lingual manner. The basic

*Corresponding author

Proceedings of NAACL-HLT 2016, pages 1041–1050,
San Diego, California, June 12-17, 2016.

idea is that the semantic contribution of Chinese characters in most Chinese words can be learned from their translations in other languages. Such as the word "青蛙" we mentioned above. The word embeddings of other languages are used to calculate semantic contribution of characters to the word they compose. Moreover, Chinese characters are more ambiguous than words. To tackle this problem, multiple-prototype character embeddings is proposed. Different meanings of characters will be represented by different embeddings. Our contributions can be summarized as follows:

1. We provide a method to calculate the semantic contribution of Chinese characters to the word they compose with English translation. Compared with English, there are fewer human-made resources to supervise the learning process of Chinese word and character embeddings. While translation resources are always easy to be accessed on the Internet.

2. We propose a novel way to disambiguate Chinese characters with translating resources. There are some limitations in existing cluster-based algorithms (Huang et al., 2012; Neelakantan et al., 2015; Chen et al., 2015). They either fixed the number of clusters or proposed a nonparametric way to learn it for each word. However, the number of clusters for words varies a lot. For nonparametric method, different hyperparameters have to be tune to control the number of clusters for different datasets.

3. We provide a method to distinguish whether a Chinese word is semantically compositional automatically. Not all Chinese words exhibit semantic compositions from their component characters. For example, entity names, transliterated words like "沙发" (sofa), single-morpheme multi-character words like "徘徊" (wander). In Chen et al. (2015), they performed part-of-speech tagging to identify entity names. The transliterated words are tagged manually, which requires human work and need to be updated when new words are created.

The evaluations on word similarity, text classification, Chinese characters disambiguation, and qualitative analysis of word embeddings demonstrate the effectiveness of our method.

2 Related Work

2.1 Word2vec

Word2vec (Mikolov et al., 2013a) is an algorithm to learn distributed word representations using a neural language model. Word2vec has two models, the continuous bag-of-words model (CBOW) and the skip-gram model. In this paper, we propose a new model based on the CBOW, hence we focus attention on it. CBOW aims at predicting the target word given context words in a slide window. Given a word sequence $D = \{x_1, x_2, \ldots, x_T\}$, the objective of CBOW is to maximize the average log probability

$$L = \frac{1}{T}\sum_{i=1}^{T} \log p(x_i | x_{i-j}^{i+j}), \tag{1}$$

where x_{i-j}^{i+j} is the context words centered at x_i, $p(x_i|x_{i-j}^{i+j})$ is defined as:

$$\frac{\exp({v'_{x_i}}^T \sum_{-j\le k\le j, k\neq 0} v_{x_{i+k}})}{\sum_{x=1}^{W} \exp({v'_x}^T \sum_{-j\le k\le j, k\neq 0} v_{x_{i+k}})}, \tag{2}$$

where v_{x_i} and v'_{x_i} are the input and output vector representations of word x_i. Since the size of English vocabulary W may be up to 10^6 scale, hierarchical softmax and negative sampling (Mikolov et al., 2013b) are applied during training to learn the model efficiently. However, using CBOW to learn Chinese word embeddings directly may have some limitations. It fails to capture the internal structure of words. In (Botha and Blunsom, 2014; Luong et al., 2013; Trask et al., 2015; Chen et al., 2015), they demonstrated the usefulness to exploit the internal structure of words, and proposed some morphological-based methods. For example, Chen et al. (2015) exploit the internal structure in Chinese words.

2.2 The CWE model

The basic idea of CWE is that both external context words and internal component characters in words provide rich information in modeling the semantic meaning of the target word. In CWE, they learned word embeddings with its component characters embeddings. Let C denotes the Chinese characters set, and the word x_t in context x_{i-j}^{i+j} is composed by

several characters in C, let $x_t = \{c_1, c_2, \ldots, c_{N_t}\}$, c_k denotes the k-th character in x_t,

$$\widehat{v}_{x_t} = v_{x_t} + \frac{1}{N_t} \sum_{k=1}^{N_t} v_{c_k}, \tag{3}$$

where $\widehat{v}_{x_t}$ is the modified word embedding, N_t denotes the number of Chinese characters in x_t. To address the issue of ambiguity in Chinese characters, they proposed several approaches for multiple-prototype character embeddings: position-based, cluster-based, nonparametric methods, and position-cluster-based character embeddings. These methods are denoted as CWE+P, CWE+L, CWE+N, CWE+LP respectively. However, this model has some limitations. The internal characters are of the same contribution to the semantic meaning of the word in CWE, which is not the case for most Chinese words.

3 Methodology

Our method can be described as three stages:

- **Obtain translations of Chinese words and characters**
  Chinese words segmentation tool is used to segment words in Chinese corpus. Then we use an online English-Chinese translation tool to translate all the Chinese characters and segmented words.

- **Perform Chinese character sense disambiguation**
  We train an English corpus with CBOW to get English word embeddings. Then, we merge some meanings of Chinese characters with small difference, and disambiguate the meanings of characters in words by computing the similarity between their English translation words.

- **Learn word and character embeddings with our model**
  Based on the character sense disambiguation process, we modify the objective of CWE to learn Chinese word and character embeddings. Then we analyse the complexity of our model briefly.

3.1 Obtain translations of Chinese words and characters

We use segmentation tools to segment words in Chinese training corpus, and perform part-of-speech tagging to recognize all the entity names. Since entity name words do not exhibit semantic compositions, they are identified as non-compositional words. We count the times of characters appearing in different words. Words with Chinese characters rarely combined with other characters are classified as single-morpheme multi-character words and identified as non-compositional.

Then programming interface of online translation tool is used to translate Chinese words and characters into English. For non-compositional Chinese words, they are not included in the translation list. Table 1 shows the English meanings of Chinese word "音乐", "沙发" and their component characters "音" and " 乐", "沙" and "发".

3.2 Perform Chinese character sense disambiguation

We train an English corpus with CBOW to get English word embeddings. Then, the meanings of characters with small difference are merged.

In Table 1, we observe that the difference between some meanings of character "乐" is very small, some of them differ only in their part-of-speech. In Chinese, the same characters and words are used in different part-of-speech but express the same semantic meaning. Hence these meanings are merged as one semantic meaning. Let $\mathrm{Sim}(\cdot)$ denotes the function to calculate the similarity between meanings of Chinese words and characters, we use cosine distance as the distance metric. The i-th and j-th meanings of Chinese character c are c^i and c^j. Their similarity is defined as:

$$\begin{aligned} &\mathrm{Sim}(c^i, c^j) = \max(cos(v_{x_m}, v_{x_n})), \\ &s.t. \quad x_m \in \mathrm{Trans}(c^i), x_n \in \mathrm{Trans}(c^j), \\ &\qquad x_m, x_n \notin \mathrm{stop\_words(en)}, \end{aligned} \tag{4}$$

where $\mathrm{Trans}(c^i)$ denotes the English translation words set of c^i, $\mathrm{stop\_words(en)}$ denotes the stop words in English, x_m and x_n are not in these stop words. For example, the Chinese word "音乐" in Table 1, c_2 denotes the second character

Word	English Explanation
音乐	music;
音	(声音) sound; (消息) news, tidings; (音质) tone; (姓氏) a surname;
乐	N. (音乐) music; (姓氏) a surname; (愉快; 满足) pleasure, enjoyment; JJ. (快乐) happy,glad,joyful,cheerful; V. (喜欢) enjoy, be glad to, love, find pleasure in; (笑) laugh, be amused; RB. (乐意) gladly, happily, willingly;
沙发	sofa, settee;
沙	N. (沙子) sand; (某些呈沙状的食物) granulated, powdered; (姓氏) a surname; JJ. (嗓音不清脆) (of voice) hoarse, husky;
发	N. (头发) hair; V. (送出;交付) send out, distribute, deliver; (发射) launch, discharge, shoot, emit; (产生, 发生) produce, generate, come into existence; (表达) express, utter; (扩大, 开展) expand, develop; (因得财物而兴旺) flourish; (放散, 散开) spread out, disperse, diffuse; etc.;

Table 1: English Translation of Chinese words and characters in ICIBA. V., N., JJ., RB. denote their verb, noun, adjective and adverb meaning respectively. Different meanings of word and character are separated by semicolon.

"乐" in the word. $\mathrm{Trans}(c_2^3)$ is the third translation English words set of character "乐", which is $\{pleasure, enjoyment\}$. Therefore x_m can be pleasure or enjoyment here.

If the $\mathrm{Sim}(c^i, c^j)$ is above a threshold δ, then they are merged as one semantic meaning. For simplicity, we use the union of English translation words set. One character may be translated into several English words. We may average all the translation word embeddings and then compute the similarity, or select the maximum value of the similarity between all English word pairs. In our experiments, maximum method works better.

Finally, we perform Chinese character sense disambiguation. In Chinese, characters may have multiple meanings, but for a certain word, their meanings are determined. For exmaple, the word "音乐", the English translation is music. For character "乐", the first translation "music" matches the meaning of the word. For character "音", the best match is the first translation "sound". For transliterated word like "沙发", the English translations are sofa and settee, neither sofa nor settee have high similarity with English translation words of character "沙" and character "发". Formally, if $\max(\mathrm{Sim}(x_t, c_k)) > \lambda, c_k \in x_t$, then x_t is identified as compositional word, and belongs to the compositional set COMP. For compositional words, we build a set

$$F = \{(x_t, s_t, n_t) \mid x_t \in \mathrm{COMP}\}, \tag{5}$$

where

$$\begin{aligned} s_t &= \{\mathrm{Sim}(x_t, c_k) \mid c_k \in x_t\}, \\ n_t &= \{\max_i \mathrm{Sim}(x_t, c_k^i) \mid c_k \in x_t\} \end{aligned} \tag{6}$$

For example, the word "音 乐" is defined as ("音 乐", {Sim("音 乐", "音"), Sim("音 乐", "乐")},{1,1}) in F.

3.3 Learn word and character vectors with SCWE

The internal characters in a word make different contributions to its semantic meaning. However, in Chen et al. (2015), the contribution of component characters to the semantic meaning of word are treated equally. They add character embeddings to the word embeddings with the same weight, which may undermine the quality of word embeddings. Based on this point, we propose a similarity-based character-enhanced word embedding model, which takes the contribution of characters into account. We name it SCWE for ease of reference in the later part. The architecture of CWE and SCWE are shown in Fig. 1.

Similarity-Based Character-Enhanced word Embedding In the character sense disambiguation stage, we build a set F, which contains compositional words, the similarity between words and its component characters, and the meaning order number of characters in the word. Suppose x_t in W is a

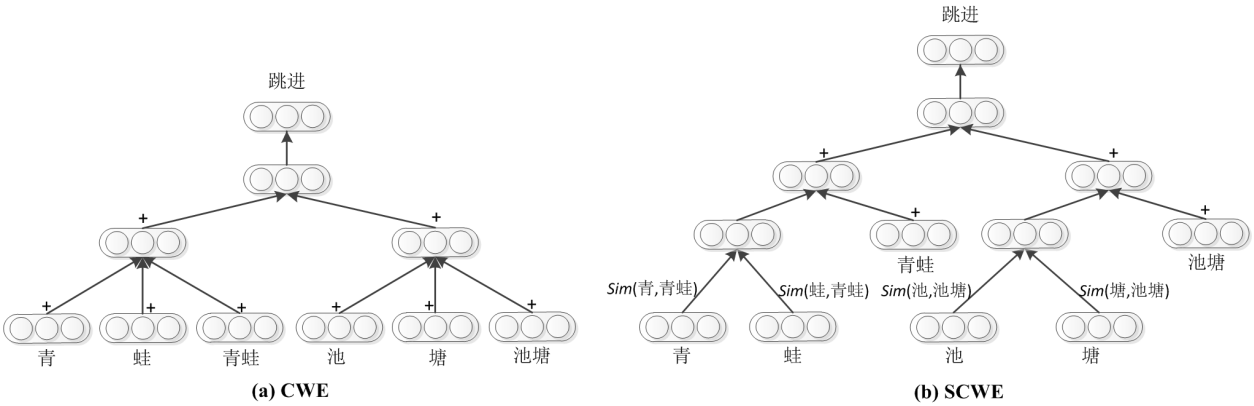

Figure 1: Architecture of models. The left is CWE and right is SCWE. "青蛙 (frog) 跳进 (jump into) 池塘 (pond)" is the word sequence. The word "青蛙" is composed of characters "青 (blue or green)" and "蛙 (frog)", and the word "池塘 (pond) is composed of characters "池 (pond, pool)" and "塘 (pond)".

compositional word, in SCWE,

$$\widehat{v}_{x_t} = \frac{1}{2}\Big\{v_{x_t} + \frac{1}{N_t}\sum_{k=1}^{N_t}\mathrm{Sim}(x_t, c_k)v_{c_k}\Big\} \quad (7)$$

To deal with ambiguity problem of Chinese characters, we propose multiple-prototype character embeddings and denote it as SCWE+M model. Since the meaning of a character is determined in a given word, we utilize the information provided by the last element in set F, and use different character embeddings for different meanings of characters. Then, in SCWE+M,

$$\widehat{v}_{x_t} = \frac{1}{2}\Big\{v_{x_t} + \frac{1}{N_t}\sum_{k=1}^{N_t}\mathrm{Sim}(x_t, c_k)v_{c_k^i}\Big\} \quad (8)$$

Complexity analysis We analyze the complexities of CBOW, CWE, SCWE and SCWE+M. Let S denotes the size of corpus, $|W|$ denotes the size of vocabulary, $|C|$ denotes the number of Chinese characters in corpus. And d is the dimensions of Chinese word and character embeddings, k is the context window size, f is the time spend in computing hierarchical softmax or negative sampling, n is the average number of characters in a Chinese word, m is the average meaning number of Chinese characters. The results are shown in Table 2.

In Chinese, most of words are composed by two Chinese characters, and the meaning number of commonly used characters are usually less than five. Moreover, according to *CJK Unified Ideographs*[1], the total number of Chinese characters is 20913, the commonly used characters are less than 10000. Therefore, our model is competitive to other methods in model parameters and computational complexity.

Method	Model parameters	Computational complexity
CBOW	$\|W\|d$	$2kSf$
CWE	$(\|W\|+\|C\|)d$	$2kS(f+n)$
SCWE	$(\|W\|+\|C\|)d$	$2kS(f+n)$
SCWE + M	$(\|W\|+m\|C\|)d$	$2kS(f+n+mn)$

Table 2: Complexity analysis

4 Experiments and Analysis

4.1 Experiments Settings

We select *English Wikipedia Dump*[2] to train English word embeddings with CBOW, and set dimensions to 200. For Chinese word embeddings, we select *Chinese Wikipedia Dump*[3] to train character and word embeddings. Before training, pure digits and non-Chinese characters are removed. We use an open-source Chinese segment tool called *ANSJ*[4] to

[1] https://en.wikipedia.org/wiki/CJK_Unified_Ideographs

[2] http://download.wikipedia.com/enwiki/

[3] http://download.wikipedia.com/zhwiki/

[4] https://github.com/NLPchina/ansj_seg

segment words in corpus. ANSJ is a java implementation of ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System). It can process about one million words in a second, and get up to 96 percent accuracy in segmentation task. The part-of-speech tagging and name entity recognition tasks are also done in this process. We select *ICIBA*[5] as English-Chinese translation tool, which provides us with an application programming interface. CBOW and CWE are used as baseline methods. Context window size is set as 5 and both Chinese word and character embeddings are set as 100 dimension. After some cross validation steps, our threshold δ and λ are set as 0.5 and 0.4 in character disambiguation process. The influence of λ and δ is report in the later part.

Model	wordsim-240	wordsim-296
CBOW	51.78	60.82
CWE	52.57	60.36
SCWE	54.92	60.85
SCWE + M	**55.10**	**62.86**

Table 3: Evaluation on wordsim-240 and wordsim-296

4.2 Word Similarity

Word similarity is a task to compute semantic relatedness between given word pairs. The relatedness between word pairs have been scored by human in advance. The correlation between model results and human judgement can be used to evaluate the performance of models. In this paper, wordsim-240 and wordsim-296 (Jin and Wu, 2012) are used as evaluation datasets. The Spearman's rank correlation (Myers et al., 2010) is applied to compute the correlation. The experimental results are summarized in Table 3.

We observe that on wordsim-240, SCWE and SCWE+M outperform the baseline methods, which indicates the effectiveness of exploiting the internal structure. On dataset wordsim-296, we can see that CBOW, CWE, SCWE perform similarly. This may be explained by some highly ambiguous Chinese characters in this dataset. In SCWE and CWE, representing these ambiguous characters with the same embeddings may undermine word embeddings. Therefore, SCWE+M achieves a better performance by applying multiple-prototype character embeddings.

Fudan-large	Size
Environment	1218
Agriculture	1022
Economy	1601
Politics	1025
Sports	1254

Fudan-small	Size
Education	59
Philosophy	44
Transport	58
Medical	52
Military	75

Table 4: 2 groups datasets of text classification, the first column denotes the category of documents and the second denotes number of documents in each category.

4.3 Text Classification

In this experiment, we use *Fudan Corpus*[6] as datasets, which contains 20 categories of documents, including economy, politics, sports and etc.. The number of documents in each category ranges from 27 to 1061. To avoid imbalance, we select 10 categories and organize them into 2 groups. One group is named *Fudan-large* and each category in this group contains more than 1000 documents. The other is named *Fudan-small* and each category contains less than 100 documents. In each category, 80 percent of documents are used as training set, the rest are used as testing set to evaluate the performance. The detailed information for two datasets are reported in Table 4.

Similar to the way we deal with Chinese training corpus, pure digits and non-Chinese characters are removed and ANSJ is used to do word segmentation on these datasets. The publish information of each document is removed. We represent each document by averaging word embeddings in the document. The classifiers are trained using LIBLINEAR package(Fan et al., 2008) with the embeddings obtained from different methods. The performance of each method is evaluated by predicting accuracy on testing set. Experiment results are given in Table 5.

It is observed that our methods outperform the baseline methods on both datasets. This can be explained that the semantic relatedness of a word with the component characters which have more contribution to its semantic meaning is strengthen in our methods. Such as, in sports documents, the word

[5] http://www.iciba.com/

[6] http://www.datatang.com/data/44139

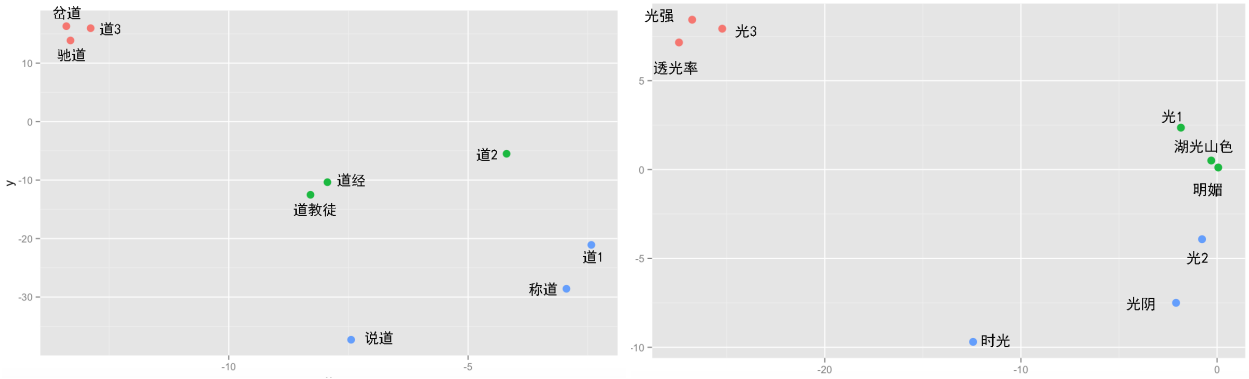

Figure 2: Illustration of words and characters in two dimension plane.

Method	Fudan-small	Fudan-large
CBOW	84.75	91.42
CWE	88.14	91.84
SCWE	91.53	92.68
SCWE + M	**93.22**	**92.89**

Table 5: Evaluation accuracies (%) on text classification.

"球" (ball) is used frequently. For Chinese words like "篮球" (basketball) and "网球" (tennis), the character "球" contributes more to their semantic meaning than other characters. Therefore, they lie closer to character "球" in embedding space obtained by our model than CBOW and CWE, and tend to form a cluster in embedding space.

Characters	Words
道1 (say, speak)	说道 (say) 称道 (speak)
道2 (Taoism, Taoist)	道经 (Taoist scriptures) 道教徒 (Taoist)
道3 (road, path)	岔道 (branch road) 驰道 (royal road)
光1 (scenery)	湖光山色 (a landscape of lakes and mountains) 明媚 (bright and beautiful)
光2 (time)	时光 (time, year) 光阴 (time)
光3 (light,ray)	光强 (light intensity) 透光率 (light transmittance)

Table 6: English explanatory of characters and their nearest words in vector space.

4.4 Multiple Prototype of Chinese Charaters

To tackle the ambiguity of Chinese characters, we propose multiple-prototype character embeddings. To evaluate the effectiveness of our method, we use PCA to conduct dimensionality reduction on word and character embeddings. The results are illustrated in Fig 2. We take 3 different meanings of Chinese characters "道" and "光", and 2 of their top-related words as examples. The character followed by a digit i denotes the i-th meanings of it.

We can observe that characters and words, which have similar meanings are gathered together. For example, "光3", "光强" and "透光率" are all related to the light. Thus, they get closer in the embedding space.

We also develop a dataset to compare our method with the disambiguation methods in Chen et al. (2015). We select some ambiguous Chinese characters, and then use *online Xinhua Dictionary*[7] as our standard to disambiguate the words that contain these ambiguous characters. Each word is assigned a number according to their explanation in the dictionary. We use KNN as classifier to evaluate all the methods. The results are shown in Table 7. It is observed that our method outperforms the methods proposed in Chen et al. (2015).

Model	Accuracy
CWE + P	84.9
CWE + L	81.0
CWE + LP	85.4
CWE + N	73.5
SCWE + M	**91.1**

Table 7: Evaluation accuracies (%) on ambiguous characters.

[7] http://xh.5156edu.com/

Words	CWE	SCWE
青蛙 (frog)	青蛇 (green snake) 青蟹 (blue crab) 青椒 (green pepper) 牛蛙 (Rana catesbeiana)	牛蛙 (Rana catesbeiana) 狐狸 (fox) 螃蟹 (crab) 蛙 (frog)
电话 (telephone)	电话网 (telephone network) 电邮 (Email) 电话卡 (phonecard) 长途电话 (toll call)	电讯 (dispatch) 手机 (cellphone) 通讯 (communication) 短信 (message)

Table 8: Nearest words example of Chinese words.

4.5 Qualitative analysis of word embeddings

In this part, we take two Chinese words as examples, and list their nearest words to examine the quality of word embeddings obtained by CWE and SCWE. The results are shown in Table 8. We can observe the most similar words return by CWE and SCWE both tend to share common characters with the given word. In CWE, characters with little semantic contribution to the word may undermine the quality of word embeddings. For example, the character "青" in word "青蛙". The semantic relatedness of words with character "青" to the given word are overestimated in CWE. In our model, by calculating the semantic contribution of internal characters to the word, we alleviate this misjudgement greatly, which demonstrates the effectiveness of our model.

4.6 Parameter Analysis

In this part, the influence of parameters on our model is investigated. The parameters include the compositional word similarity threshold λ, character disambiguation threshold δ.

Compositional word similarity To investigate how λ influence the process of non-compositional word detection, we build a word list of transliterated words manually, which consists of 161 words. Then 161 of most frequent semantic compositional words with more than one Chinese characters are added to the list in the corpus. In Table 9, the performance of our method in classifying transliterated words when λ ranges from 0.25 to 0.55 are reported. From Table 9, we can observe as λ increases, more compositional words will be classified as non-compositional words, while transliterated words are more likely to be classified correctly. Our method achieves best F-Score when $\lambda = 0.4$.

Character disambiguation threshold In Table 10, we show the performance of our model in disambiguating Chinese characters. We adopted the same datasets in Section 4.4 with different δ. From Table 1, we can observe some meanings of a character are very close, therefore, a high δ are adopted in our model. When $\delta = 0.5$, our model gets the best result in our dataset.

Parameter λ	Precision	Recall	F-Score
0.25	97.0	60.9	74.8
0.30	96.5	68.9	80.4
0.35	94.6	75.8	84.2
0.40	92.0	78.9	**85.0**
0.45	88.9	80.1	84.3
0.50	84.0	84.5	84.2
0.55	82.5	85.1	83.8

Table 9: Precision, recall, F-score of transliterated words when λ ranges from 0.25 to 0.55

Parameter δ	Precision
0.35	87.5
0.40	89.0
0.45	89.5
0.50	**91.1**
0.55	89.9
0.60	89.5
0.65	88.5

Table 10: Evaluation accuracies (%) on ambiguous characters when λ ranges from 0.35 to 0.65.

5 Conclusion

In this paper, we exploit the internal structure in Chinese words by learning the semantic contribution of internal characters to the word. We propose a method to improve Chinese word and character embeddings with a similarity-based character-enhanced word embeddings model. Ambiguity problem of Chinese characters can also be tackled in our method. Moreover, we build a way to classify whether a Chinese word is compositional

automatically, which requires to be labelled manually in CWE. We argue that our method may be used to improve word embeddings of other language whose internal structure is similar to Chinese. The code and datasets we use is available at: `https://github.com/JianXu123/SCWE`.

Acknowledgement

This work is supported by NSFC grants 91546116 and 61511130083.

References

[Bengio et al.2006] Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin, and Jean-Luc Gauvain. 2006. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. Springer.

[Botha and Blunsom2014] Jan A Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. *arXiv preprint arXiv:1405.4273*.

[Chen et al.2015] Xinxiong Chen, Lei Xu, Zhiyuan Liu, Maosong Sun, and Huanbo Luan. 2015. Joint learning of character and word embeddings. *In Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*.

[Fan et al.2008] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

[Hermann and Blunsom2014] Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. *arXiv preprint arXiv:1404.4641*.

[Huang et al.2012] Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng. 2012. Improving word representations via global context and multiple word prototypes. In *In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 873–882. Association for Computational Linguistics.

[Jin and Wu2012] Peng Jin and Yunfang Wu. 2012. Semeval-2012 task 4: evaluating chinese word similarity. In *In Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 374–377. Association for Computational Linguistics.

[Kalchbrenner et al.2014] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

[Kiros et al.2015] Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in Neural Information Processing Systems*, pages 3276–3284.

[Le and Mikolov2014] Quoc V Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053*.

[Li et al.2015] Yanran Li, Wenjie Li, Fei Sun, and Sujian Li. 2015. Component-enhanced chinese character embeddings. *arXiv preprint arXiv:1508.06669*.

[Luong et al.2013] Minh-Thang Luong, Richard Socher, and Christopher D Manning. 2013. Better word representations with recursive neural networks for morphology. *CoNLL-2013*, 104.

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

[Mikolov et al.2013b] 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.

[Mnih and Hinton2009] Andriy Mnih and Geoffrey E Hinton. 2009. A scalable hierarchical distributed language model. In *Advances in neural information processing systems*, pages 1081–1088.

[Myers et al.2010] Jerome L Myers, Arnold Well, and Robert Frederick Lorch. 2010. *Research design and statistical analysis*. Routledge.

[Neelakantan et al.2015] Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2015. Efficient non-parametric estimation of multiple embeddings per word in vector space. *arXiv preprint arXiv:1504.06654*.

[Rumelhart et al.1988] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1988. Learning representations by back-propagating errors. *Cognitive modeling*, 5:3.

[Socher et al.2010] Richard Socher, Christopher D Manning, and Andrew Y Ng. 2010. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *In Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*, pages 1–9.

[Socher et al.2013] Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *In Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642. Citeseer.

[Srivastava et al.2013] Nitish Srivastava, Ruslan R Salakhutdinov, and Geoffrey E Hinton. 2013. Modeling documents with deep boltzmann machines. *arXiv preprint arXiv:1309.6865*.

[Trask et al.2015] Andrew Trask, David Gilmore, and Matthew Russell. 2015. Modeling order in neural word embeddings at scale. *arXiv preprint arXiv:1506.02338*.

[Yu and Dredze2015] Mo Yu and Mark Dredze. 2015. Learning composition models for phrase embeddings. *Transactions of the Association for Computational Linguistics*, 3:227–242.

[Zhang et al.2014] Jiajun Zhang, Shujie Liu, Mu Li, Ming Zhou, and Chengqing Zong. 2014. Bilingually-constrained phrase embeddings for machine translation. In *In Proceedings of the 52th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics*.