A Hybrid Learning Scheme for Chinese Word Embedding

Wenfan Chen[1] and **Weiguo Sheng**[2,*]

[1] School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, P.R.China
[2] Department of Computer Science, Hangzhou Normal University, Hangzhou, P.R.China

Abstract

To improve word embedding, subword information has been widely employed in state-of-the-art methods. These methods can be classified to either compositional or predictive models. In this paper, we propose a hybrid learning scheme, which integrates compositional and predictive model for word embedding. Such a scheme can take advantage of both models, thus effectively learning word embedding. The proposed scheme has been applied to learn word representation on Chinese. Our results show that the proposed scheme can significantly improve the performance of word embedding in terms of analogical reasoning and is robust to the size of training data.

1 Introduction

Word embedding, also known as distributed word representation, represents a word as a real-valued low-dimensional vector and encodes its semantic meaning into the vector. It is a fundamental task of natural language processing (NLP), such as language modeling (Bengio et al., 2003; Mnih and Hinton, 2009), machine translation (Bahdanau et al., 2014; Sutskever et al., 2014), caption generation (Xu et al., 2015; Devlin et al., 2015) and question answering (Hermann et al., 2015).

Most previous word embedding methods suffer from high computational complexity and have difficulty to be applied to large-scale corpora. Recently, Continuous Bag-Of-Words (CBOW) and Skip-Gram (SG) models (Mikolov et al., 2013a), which can alleviate the above issue, have received much attention. However, these models take a word as a basic unit but ignore rich subword information, which could significantly limit their performance. To improve the performance of word embedding, subword information, such as morphemes and character *n*-grams, has been employed (Luong et al., 2013; Qiu et al., 2014; Cao and Rei, 2016; Sun et al., 2016a; Wieting et al., 2016; Bojanowski et al., 2017). While these methods are effective, they are originally developed for alphabetic writing systems and can't be applied directly to other writing systems, like Chinese.

In Chinese, each word typically consists of less characters than in English[1], while each character can have a complicated structure of its meaning. Typically, a Chinese character can be decomposed into components (部), where each component has its own meaning. The internal semantic meaning of a Chinese word emerges from such a structure. For example, the Chinese word "海水 (seawater)" is composed by "海 (sea)" and "水 (water)". The semantic component of "海 (sea)" is "氵", which is the transformation of "水 (water)" and indicates it is related to "水 (water)". Therefore, the word "海水 (seawater)" has the meaning of "water from the sea".

Based on the linguistic feature of Chinese, recent methods have used subword information to improve Chinese word embedding. For example, Chen et al. (2015) proposed a character-enhanced word embedding (CWE) model, which departed from CBOW of representing context words with both character embeddings and word embeddings. Shi et al. (2015) proposed a radical embedding method, which used the CBOW framework but replacing word embeddings with radical embeddings. Yin et al. (2016) and Xu et al. (2016) extended the CWE model in different ways: the former presented a multi-granularity embedding (MGE) model, additionally using the embeddings associated with radicals detected in the target word; the latter proposed a similarity-based character-enhanced word embedding (SCWE) model, considering the similarity between a word and its

* Corresponding author. E-mail: w.sheng@ieee.org

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

Proceedings of the 3rd Workshop on Representation Learning for NLP, pages 84–90
Melbourne, Australia, July 20, 2018.

component characters. Yu et al. (2017) introduced a joint learning word embedding (JWE) model, which jointly learned embeddings for words, characters and components, and predicted the target word, respectively. Cao et al. (2018), on the other hand, represented Chinese words as sequences of strokes[2] and learned word embedding with stroke *n*-grams information.

The above methods can be divided into two types: compositional and predictive model. The compositional model composes rich information into one vector to predict the target word. In this type of model, information works in a cooperative manner for word embedding. By contrast, the predictive model decouples various information to predict the target word. The information in this type of model works competitively for word embedding. Both models can effectively learn word embedding and give good estimation for rare and unseen words. By combining richer information, the compositional model can more accurately represent the target word. However, information is usually composed in a sophisticated way. The predictive model, on the other hand, is simple and can directly capture the interaction between words and their internal information. This type of model, however, typically ignores the interrelationship between various information.

To take advantage of both models, in this paper, we propose a hybrid learning scheme for word embedding. The proposed scheme learns word embedding in a competitive and cooperative manner. Specifically, in our scheme, the decoupled representations are used to capture the semantic meaning of target word respectively while making their composition semantically consistent with the target word. The performance of proposed scheme has been evaluated on Chinese in terms of word similarity and analogy tasks. The results show that our proposed scheme can effectively learn word representation and is robust to the size of training data.

2 Proposed Scheme

In this section, we present the details of our proposed hybrid learning scheme for word embedding. We denote the proposed scheme as **C**o-**O**petition **W**ord **E**mbedding (COWE). It consists of predictive and compositional parts, which will be described in subsection 2.1 and subsection 2.2,

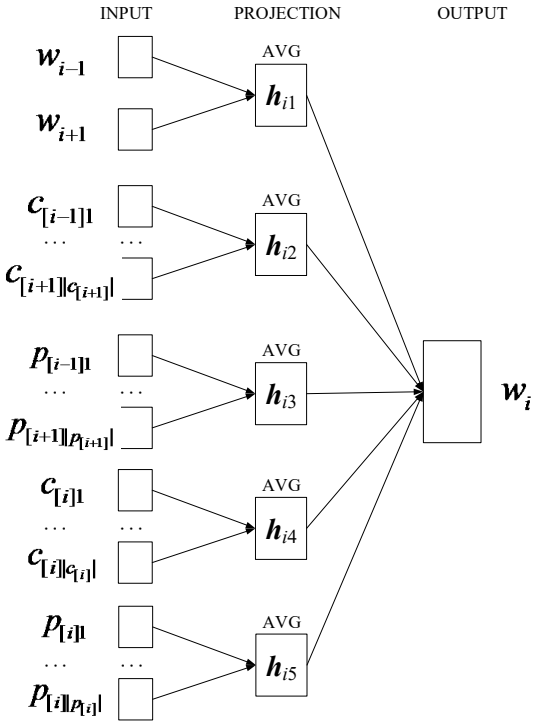

Figure 1: Illustration of the predictive part of COWE.

respectively. This is followed by describing the objective function.

The meaning of notation used in this section is as follows. We denote the training corpus as $\mathcal{D}$, word vocabulary as $\mathcal{W}$, character vocabulary as $\mathcal{C}$, components vocabulary as $\mathcal{P}$. Each word $w \in \mathcal{W}$, character $c \in \mathcal{C}$ and component $p \in \mathcal{P}$ are associated with vectors $\boldsymbol{w} \in \mathbb{R}^d$, $\boldsymbol{c} \in \mathbb{R}^d$, $\boldsymbol{p} \in \mathbb{R}^d$, respectively, where d is the vector dimension. The characters and components in word w_i are denoted as $c_{[i]}$ and $p_{[i]}$, where $|c_{[i]}|$ and $|p_{[i]}|$ denote the number of characters and components in w_i, respectively.

2.1 Predictive Part

In the predictive part, the compositions of context words, characters and components as well as compositions of characters and components in target word are used to predict the target word, as illustrated in Figure 1. These separate predictions by various compositions can be considered as competitions for the semantic meaning of target word. In order to maintain similar length between different compositions, COWE uses an average operation as the composition operation.

[2] https://en.wikipedia.org/wiki/Stroke_(CJKV_character)

The goal of this part is to maximize the sum of log likelihoods of all predictive conditional probabilities:

$$\mathcal{L}_p(w_i) = \sum_{k=1}^{5} \log p(\boldsymbol{w}_i|\boldsymbol{h}_{ik}), \quad (1)$$

where $\boldsymbol{h}_{i1}$, $\boldsymbol{h}_{i2}$, $\boldsymbol{h}_{i3}$, $\boldsymbol{h}_{i4}$ and $\boldsymbol{h}_{i5}$ correspond to the above mentioned five compositions, respectively. Here, $\boldsymbol{h}_{i1}$ is defined as:

$$\boldsymbol{h}_{i1} = \frac{1}{2N} \sum_{-N \le j \le N, j \ne 0} \boldsymbol{w}_{i+j}, \quad (2)$$

where N is the context window size. $\boldsymbol{h}_{i2}$, $\boldsymbol{h}_{i3}$, $\boldsymbol{h}_{i4}$ and $\boldsymbol{h}_{i5}$ are defined in a similar way. The conditional probability is defined using a softmax function as:

$$p(\boldsymbol{w}_i|\boldsymbol{h}_{ik}) = \frac{\exp(\boldsymbol{w}_i \cdot \boldsymbol{h}_{ik})}{\sum_{w_{i'} \in \mathcal{W}} \exp(\boldsymbol{w}_{i'} \cdot \boldsymbol{h}_{ik})}, \; k = 1,2,3,4,5. \quad (3)$$

This objective function is similar to the one used in JWE (Yu et al., 2017). The main difference is that we further decouple components in the context words and target word, and leverage characters in the target word in addition.

2.2 Compositional Part

In the compositional part, all compositions mentioned above work in a cooperative manner, where their composition is used to predict the target word. We consider the composition as *semantic consistency point* of various representations, and the prediction loss as *consistency loss*, as shown in Figure 2.

The goal of this part is to maximize the following objective function:

$$\mathcal{L}_c(w_i) = \log p(\boldsymbol{w}_i|\boldsymbol{a}_i), \quad (4)$$

where $\boldsymbol{a}_i$ is the semantic consistency point, and is defined as:

$$\boldsymbol{a}_i = \frac{1}{5} \sum_{k=1}^{5} \boldsymbol{h}_{ik}. \quad (5)$$

Similar to the predictive part, the conditional probability is defined using the softmax function (see Equation (3)).

2.3 Objective Function

As COWE consists of predictive and compositional parts, its objective function is therefore consisted of the sum of all prediction losses and the consistency loss:

$$\mathcal{L}(\mathcal{D}) = \sum_{w_i \in \mathcal{D}} \mathcal{L}_p(w_i) + \mathcal{L}_c(w_i). \quad (6)$$

To solve the above optimization problem, we employ the negative sampling technique (Mikolov et al., 2013b). Note that only the consistency loss between semantic consistency point and target word is considered. In preliminary experiments, we also tried the consistency losses between semantic consistency point and sampled negative words, but observed reduced performance.

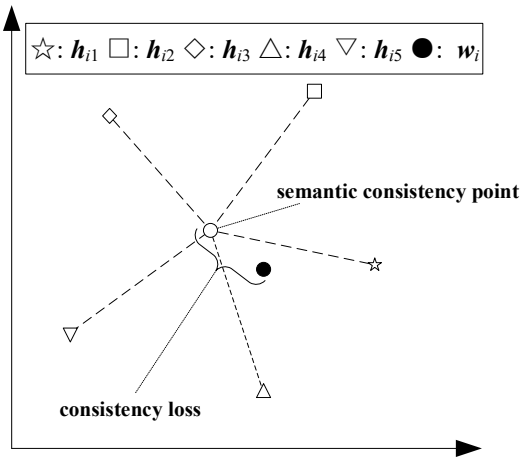

Figure 2: Illustration of the semantic consistency point and the consistency loss.

As a result, the final objective function can be written as:

$$\mathcal{L}(\mathcal{D}) = \sum_{w_i \in \mathcal{D}} \sum_{k=1}^{5} \log \sigma(\boldsymbol{w}_i \cdot \boldsymbol{h}_{ik}) + \lambda \mathbb{E}_{\widetilde{w} \sim P_{\widetilde{\mathcal{W}}}} [\sum_{k=1}^{5} \log \sigma(\widetilde{\boldsymbol{w}} \cdot \boldsymbol{h}_{ik})] + \log \sigma(\boldsymbol{w}_i \cdot \boldsymbol{a}_i), \quad (7)$$

where σ is a sigmoid function: $\sigma(\mathrm{x}) = 1/(1 + \exp(-\mathrm{x}))$, λ is the number of negative words, $\widetilde{w}$ is the sampled negative word and $P_{\widetilde{\mathcal{W}}}$ is the distribution of negative words.

3 Experiments

In this section, we evaluate COWE on Chinese in terms of word similarity computation and analogical reasoning.

3.1 Experimental Settings

We use Chinese Wikipedia dump dated on March 1, 2018[3] for embedding learning, which contains 310K Chinese Wikipedia articles. The data is preprocessed as follows. Firstly, construct training corpus from the Wikipedia dump with `WikiCorpus` in the `gensim` toolkit[4]. Secondly, convert traditional Chinese characters to simplified Chinese characters with the `opencc` toolkit[5]. Thirdly, remove all non-Chinese characters and Chinese words whose frequencies are less than 10

[3] https://dumps.wikimedia.org/zhwiki/20180301/
[4] https://radimrehurek.com/gensim/corpora/wikicorpus.html
[5] https://github.com/BYVoid/OpenCC

in the corpus. Finally, perform Chinese word segmentation with THULAC[6] (Sun et al., 2016b). In addition, we perform POS tagging on the training corpus using THULAC and identify all entity names for CWE (Chen et al., 2015), as it does not use the character information for non-compositional words. We use the subword files provided by Yu et al. (2017). As a result, we obtain a 1 GB training corpus with 165,507,601 words, 368,408 unique words, 20,885 unique characters and 13,232 unique components.

We compare COWE with CBOW (Mikolov et al., 2013a)[7], CWE (Chen et al., 2015)[8] and JWE (Yu et al., 2017)[9]. To further evaluate the effect of consistency loss and components, we create two variants of COWE, denoted as COWE-c2 and COWE-p. The former is indeed the JWE model with an additional consistency loss, while the latter is COWE without using component information. The same parameter settings are used for all models. Specifically, the vector dimension is set to 200, the training iteration is set to 100, both the size of context window and number of negative samples are set to 5, the initial learning rate is set to 0.025, and the subsampling threshold is set to 10^{-4}.

3.2 Word Similarity

This task is to evaluate the effectiveness of word embedding in capturing semantic similarity of word pairs. Following Yu et al. (2017), we adopt wordsim-240 and wordsim-296 datasets (Jin and Wu, 2012). Both datasets contain manually-annotated similarity scores for word pairs. In wordsim-240, words in 234 pairs appear in the training corpus, and in wordsim-296, words in 286 pairs appear in the training corpus. Unseen words are removed. The performance of word embedding is evaluated by ranking the pairs according to their cosine similarity and measuring the Spearman correlation ρ with human ratings. The results are shown in Table 1.

The results, on the wordsim-240 dataset, show that CWE performs better than CBOW, but outperformed by all other models. This could indicate the benefits of using rich information. COWE-c2 is not so good as JWE, COWE-p and COWE perform even worse. This suggests that the introduction of consistency loss, to some extent, may limit the performance of word representation. This may be due to the fact that our average semantic consistency point considers the contributions of various representations equally. With the evolution of history, however, meanings of some Chinese characters or components have degraded, making them less expressive. We plan to investigate the composition operation further in future work.

Model	wordsim-240	wordsim-296
CBOW	0.4861	0.5658
CWE	0.5151	0.5684
JWE	**0.5496**	**0.6355**
COWE-c2	0.5473	0.5899
COWE-p	0.5180	0.5844
COWE	0.5412	0.5674

Table 1: Results on word similarity evaluation.

3.3 Word Analogy

This task is to evaluate the effectiveness of word embedding in capturing semantic relations between pairs of words. The goal is to answer the analogy questions of the form "a is to a* as b is to b*", where b* is hidden, and must be reasoned out from the vocabulary. We use the Chinese word analogy dataset provided by Chen et al. (2015). It consists of 1,124 analogy questions, categorized into 3 types: 1) capitals of countries (677 groups), 2) capitals of provinces/states (175 groups), and 3) family relationships (272 groups). The analogy questions are answered using 3CosAdd (Mikolov et al., 2013a) as well as 3CosMul (Levy and Goldberg, 2014)[10]. We abbreviate the two methods as "Add" and "Mul", respectively. The evaluation metric for this task is the percentage of questions for which the argmax result is the correct answer b*. The results are shown in Table 2[11].

It can be found that CBOW performs better than CWE and JWE on the Capital and Family tasks. This is due to that using internal information improperly could be harmful in cases where words are non-compositional or irrelevant words sharing similar internal structures. For example, the words "儿子 (son)" and "妻子 (wife)" share the same character "子", which means "son" in the former but makes no sense in the latter. We observe that COWE-c2 achieves the best results

[6] http://thulac.thunlp.org/
[7] https://code.google.com/archive/p/word2vec/
[8] https://github.com/Leonard-Xu/CWE
[9] https://github.com/HKUST-KnowComp/JWE
[10] https://bitbucket.org/omerlevy/hyperwords
[11] The results do not agree with that reported in (Yu et al., 2017). We suggest that these discrepancies stem from differences in training corpus and parameter settings.

Model	Capital	State	Family
	Add/Mul	Add/Mul	Add/Mul
CBOW	87.00/85.82	93.14/92.00	76.84/73.90
CWE	86.71/85.08	91.43/90.29	75.74/70.96
JWE	86.12/83.90	**94.29**/**94.29**	70.96/69.49
COWE-c2	83.16/83.31	90.29/86.29	**77.94**/**74.63**
COWE-p	**87.74**/85.82	92.57/**94.29**	73.16/69.85
COWE	85.82/**86.12**	**94.29**/93.71	76.10/74.26

Table 2: Results on word analogy evaluation.

on the Family task and outperforms JWE by large margins. This shows the effectiveness of consistency loss in helping with learning from various information. COWE-p and COWE perform best on the other tasks, respectively. The fact suggests that different information could help in different ways.

3.4 Performance on Low-Resource Corpora

To evaluate the performance of different models on low-resource corpora, we conduct the same experiments on 5%, 10% and 20% randomly selected Wikipedia articles, respectively. As less training data introducing more noises, this makes it more difficult for models to learn good word representations. The results are shown in Table 3.

The results indicate the superiority of our models on low-resource corpora. We observe that as the size of dataset decreases, the performance of baselines drops rapidly, while the performance decrement of COWE and its variants is much smaller. This shows the robustness of our proposed models. COWE-p is generally more robust than COWE-c2, however, COWE-c2 performs more robustly on the Family task. Taking both characters and components into account, COWE achieves the most robust results.

We also observe that on the Capital task, the performance of CWE and JWE drops more quick than CBOW, which agrees with the previous findings. However, with the consistency loss, COWE-c2 always performs better than JWE, and usually outperforms CBOW. We believe that the consistency loss, in cases where some embeddings are useless, would encourage weak embeddings to close to strong embeddings, letting weak embeddings acquire some helpful features, and prevent strong embeddings from overfitting. On the State and Family tasks, where the character and component embeddings could be useful, all of our models still outperform the baselines by large margins. This should be due to the fact that the consistency loss prevents various learned embeddings from contradicting each other, thus making all of them close to the true target word embedding.

Model	Capital	State	Family
	Add/Mul	Add/Mul	Add/Mul
CBOW	57.46/52.14	28.00/23.43	34.19/29.04
CWE	51.99/47.12	36.00/31.43	16.18/13.60
JWE	44.76/40.77	49.14/44.57	31.99/27.57
COWE-c2	61.74/58.64	67.43/65.14	**44.49**/35.29
COWE-p	**79.17**/77.40	80.00/81.71	37.87/37.50
COWE	78.14/**79.03**	**81.71**/**82.86**	41.91/**41.18**

(a) 5% Wikipedia articles

Model	Capital	State	Family
	Add/Mul	Add/Mul	Add/Mul
CBOW	73.12/69.42	54.29/50.29	48.90/43.38
CWE	66.03/64.40	54.29/53.14	39.71/37.13
JWE	63.81/63.22	62.86/58.29	40.07/36.40
COWE-c2	70.16/67.50	77.71/73.14	59.19/56.62
COWE-p	77.70/**78.58**	79.43/**80.00**	54.78/52.21
COWE	**78.43**/**78.58**	**80.00**/77.71	**60.29**/**56.99**

(b) 10% Wikipedia articles

Model	Capital	State	Family
	Add/Mul	Add/Mul	Add/Mul
CBOW	70.75/68.39	69.71/64.57	59.19/54.78
CWE	67.80/65.58	66.29/63.43	50.00/44.49
JWE	70.46/69.28	81.71/78.86	48.90/48.16
COWE-c2	74.89/72.97	**90.29**/**87.43**	**59.93**/**56.25**
COWE-p	81.83/81.83	89.71/86.86	58.46/54.78
COWE	**84.79**/**83.60**	87.43/86.86	58.46/55.51

(c) 20% Wikipedia articles

Table 3: Results on word analogy evaluation, trained on 5%/10%20% Wikipedia articles.

3.5 Case Study

To gain a better understanding of the quality of learned word embedding, we take the word "癌症 (cancer)" as an example and show its nearest neighbors in Table 4, where cosine similarity is used as the distance metric.

All words yielded by different models are disease-related. Specifically, words yielded by CWE contain the character "癌 (cancer)", including some weird words, like "国家癌症 (national cancer)" and "抑癌 (anti-cancer)"[12]. This implies that CWE has overused the internal information. For

[12] Translation by Google Translate.

CWE	JWE	COWE
肾癌 (renal cr)	肺癌 (lung cr)	肺癌 (lung cr)
癌病 (cr)	肝癌 (liver cr)	并发症 (complication)
肺癌 (lung cr)	癌 (cr)	癌 (cr)
胰腺癌 (pancreatic cr)	胃癌 (gastric cr)	白血病 (leukemia)
国家癌症 (national cr)	白血病 (leukemia)	乳腺癌 (breast cr)
抑癌 (anti-cr)	肺结核 (tuberculosis)	胃癌 (gastric cr)
肝癌 (liver cr)	胰腺癌 (pancreatic cr)	肝癌 (liver cr)
脑癌 (brain cr)	肺炎 (pneumonia)	肺结核 (tuberculosis)
胰脏腺癌 (pancreatic cr)	心脏病 (heart disease)	大肠癌 (colorectal cr)
癌 (cr)	并发症 (complication)	肺炎 (pneumonia)

Table 4: Nearest neighbors of "癌症 (cancer)". "cr" is abbreviation for "cancer".

JWE and COWE, which directly capture the interaction between the words and their internal information, they yield disease-related words that do not contain the component "疒", such as "肺结核 (pneumonia)". This indicates that they make full use of external and internal information, and avoid the above issue. Compared to JWE, COWE yields more words that are semantically relevant to the target word.

4 Conclusion

This paper proposes a scheme, which combines predictive and compositional models to jointly learn various word representations in a competitive and cooperative manner. The predictive part of the proposed scheme is based on various external and internal information, which is used to capture corresponding representation. In the compositional part, the semantic consistency point and the consistency loss are introduced. They connect separate learned representations and prevent them from contradicting each other. The experimental results show that the proposed scheme outperforms baseline models on word analogy tasks and achieves competitive results on word similarity tasks. The results also show that our model is robust to the size of training data. Therefore, our proposed scheme is suitable to be applied on low-resource corpora, for example task-specific corpora, where data is often very scarce.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant Nos. 61573316). We also thank Yuehua Wan and Xiaoxu Wu for their help and very valuable feedback.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv:1409.0473*:1–15, September.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A Neural Probabilistic Language Model. *The Journal of Machine Learning Research*, 3:1137–1155.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5(1):135–146. http://aclweb.org/anthology/Q17-1010

Kris Cao and Marek Rei. 2016. A Joint Model for Word Embedding and Word Morphology. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 18–26. Association for Computational Linguistics. https://doi.org/10.18653/v1/W16-1603

Shaosheng Cao, Wei Lu, Jun Zhou, and Xiaolong Li. 2018. cw2vec: Learning Chinese Word Embeddings with Stroke n-gram Information. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pages 1–8.

Xinxiong Chen, Lei Xu, Zhiyuan Liu, Maosong Sun, and Huanbo Luan. 2015. Joint learning of character and word embeddings. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 1236–1242.

Jacob Devlin, Hao Cheng, Hao Fang, Saurabh Gupta, Li Deng, Xiaodong He, Geoffrey Zweig, and Margaret Mitchell. 2015. Language Models for Image Captioning: The Quirks and What Works. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 100–105, A Neural Probabilistic Language Model. Association for Computational Linguistics. https://doi.org/10.3115/v1/P15-2017

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching Machines to Read and Comprehend. In *Proceedings of the 28th International Conference*

on Neural Information Processing Systems, pages 1693–1701. June.

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. http://www.aclweb.org/anthology/S12-1049

Omer Levy and Yoav Goldberg. 2014. Linguistic Regularities in Sparse and Explicit Word Representations. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 171–180. Association for Computational Linguistics. https://doi.org/10.3115/v1/W14-1618

Minh-Thang Luong, Richard Socher, and Christopher D. Manning. 2013. Better Word Representations with Recursive Neural Networks for Morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113. http://www.aclweb.org/anthology/W13-3512

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*:1–12, January.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed Representations of Words and Phrases and their Compositionality. *Advances in neural information processing systems*:3111–3119, October.

Andriy Mnih and Geoffrey E. Hinton. 2009. A Scalable Hierarchical Distributed Language Model. In *Advances in neural information processing systems*, pages 1081–1088.

Siyu Qiu, Qing Cui, Jiang Bian, Bin Gao, and Tie-Yan Liu. 2014. Co-learning of Word Representations and Morpheme Representations. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 141–150. Dublin City University and Association for Computational Linguistics. https://www.aclweb.org/anthology/C14-1015

Xinlei Shi, Junjie Zhai, Xudong Yang, Zehua Xie, and Chao Liu. 2015. Radical Embedding: Delving Deeper to Chinese Radicals. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 594–598. Association for Computational Linguistics. https://doi.org/10.3115/v1/P15-2098

Fei Sun, Jiafeng Guo, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2016a. Inside Out: Two Jointly Predictive Models for Word Representations and Phrase Representations. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2821–2827.

Maosong Sun, Xinxiong Chen, Kaixu Zhang, Zhipeng Guo, Zhiyuan Liu. 2016b. THULAC: An Efficient Lexical Analyzer for Chinese.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pages 3104–3112. September.

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. https://doi.org/10.18653/v1/D16-1157

Jian Xu, Jiawei Liu, Liangang Zhang, Zhengyu Li, and Huanhuan Chen. 2016. Improve Chinese Word Embeddings by Exploiting Internal Structure. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1041–1050. Association for Computational Linguistics. https://doi.org/10.18653/v1/N16-1119

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *International Conference on Machine Learning*, pages 2048–2057.

Rongchao Yin, Quan Wang, Peng Li, Rui Li, and Bin Wang. 2016. Multi-Granularity Chinese Word Embedding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 981–986. Association for Computational Linguistics. https://doi.org/10.18653/v1/D16-1100

Jinxing Yu, Xun Jian, Hao Xin, and Yangqiu Song. 2017. Joint Embeddings of Chinese Words, Characters, and Fine-grained Subcharacter Components. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 286–291. Association for Computational Linguistics. https://doi.org/10.18653/v1/D17-1027