Learning New Word Semantics with Conceptual Text

Wei Pan
School of Software
Shandong University
Jinan, China
panwei_sdu@163.com

Tianyuan Liu
School of Software
Shandong University
Jinan, China
zodiacg@foxmail.com

Wentao Zhang
School of Computer Science and Engineering
Beihang University
Beijing, China
zhangwt@act.buaa.edu.cn

Yuqing Sun*
School of Software
Shandong University
Jinan, China
sun_yuqing@sdu.edu.cn

2021 International Joint Conference on Neural Networks (IJCNN) | 978-1-6654-3900-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/IJCNN52387.2021.9533503

***Abstract*—In this paper, we consider the embedding problem of Chinese new word with respect to its conceptual definition or description, which is especially important for understanding specialty documents. We present a two-stage model to learn the Chinese new word embedding, where the first encodes the information of character components and context, and the second aggregates the semantics of multiple texts. We perform extensive experiments to verify the proposed method and the results outperform the state of art methods on both direct semantics verification and advanced NLP tasks. Comparing with previous methods that require a corpus or an elaborately designed dataset for learning a new word embedding, our method requires only a few pieces of text and supports the evolution of meanings. We also experimentally verify the effects of different parts of model, the number and types of conceptual texts. Finally, we present some biology texts to illustrate whether the specialty semantics are encoded in the word embedding.**

***Index Terms*—Conceptual Text, Word Embedding, Aggregation**

I. Introduction

Word embeddings are defined as the quantification of distributed attributes in a dense linguistic space [1], which are often learned by the relationships between a target word and its appearance contexts. The introduction of word embedding in deep network has greatly improved the performance of most natural language tasks, such as text classification [2], named entity recognition [3], machine translation [4] and question answering [5].

The current mainstream method is to collect a large corpus to train the embeddings for words. Since the occurrences of most words in vocabulary are large enough, it can get a good result. However, the mainstream method is not applicable for the specialty words in a new area, which resides on two aspects: the low frequency of these words in a common corpus and the specialized semantics are less contained in the corpus. Specialty words refer to the unified representation of some specific things in a specific field, such as the "Indomethacin" in the medical field. According to the MIT statistics [6], the ratio of occurrences of out-of-vocabulary(OOV) words is close to 10%. This is a natural phenomenan of language in the progress of science, technology, culture etc. Since specialty words often have an important role in a sentence, i.e. keywords, the missing of word embeddings may hinder the understanding of these professional documents.

To solve this problem, some works use the component information of a word to learn the embeddings. Pinter et al. [7] proposed the mimic method to represent a word by its subword embeddings and show that with these embeddings the performances on some natural language tasks are better than with value <unk> on unknown words. For Chinese, Zhang et al. [8] integrated the features of stroke, structure and pinyin for word embeddings, which also improved the performances of some NLP tasks. These methods are not appropriate for the specialty words whose semantics are not contained in the word components. Another type of methods adopt several pieces of text or an elaborately designed data set to learn low-frequency word embeddings. Hu et al. [9] used the attention-based hierarchical context model to learn embeddings. Schick et al. designed a data set of keyword pairs. Each pair is obtained from the relation in WordNet, such as <bike> is a <bicycle>. They presented the one-token approximation (OTA) method [10] to obtain a multi-token word embedding and conclude that the integration of components and context can better capture the semantics of rare words. However, the usage of a specialty word are different with that of usual word such that their embeddings can not be learned as normal.

To tackle these issues, we consider the embedding problem of Chinese new words with conceptual texts. Since conceptual text is the exact description of the essential characteristics of a thing or the connotation of an attribute. Compared with other methods, it does not require a larger copus for training, which is efficient and especially imporatant for a new professional field. In addition, we also use the forward attention that conforms to the Chinese expression and propose an aggregation method to support an incremental update of embeddings. The contributions are summarized as follows:

- We present a two-stage model to learn a word embedding. The first is a dual attention model to predict an OOV word embedding based on a conceptual text, which integrates the contextual and structural information. Since a word semantics are related to the position and role that undertakes in a sentence, we add the part-of-speech and location information to assist encode the context by the self-attention layer. The forward attention layer helps

*Corresponding author: sun_yuqing@sdu.edu.cn

978-0-7381-3366-9/21/$31.00 ©2021 IEEE

encode the sequential usage of Chinese expression.

- Additionally, the semantics of a new word evolve with the usages and the conceptual text is often in the form of a few pieces. To support an incremental update of embeddings, the second part of our model aggregates the semantics of multiple texts. This is especially important for the new words with a few shots and is applicable in practice.
- We perform extensive experiments on six datasets to verify the effectiveness of our model. The direct verification by similarity task focuses on the semantics of the learned embeddings and the advanced NLP tasks further justify whether they benefit the succeeding tasks. Then we examine how different parts of model, the number and types of text influence the results. Besides, we choose Biology as the specialty domain to verify whether characteristics of new words are encoded in embeddings. The source code and dataset of this paper can be obtained on Github. [1]

The rest of this paper is organized as follows. Sec. II introduces related works. In Sec. III we present the proposed embedding method. Sec. IV introduces the datasets and the experimental results. Finally, we conclude this paper.

II. Related Work

The concept of distributed representation of words was coined by Rumelhart [11] that reflect the semantic correlations between words in a continuously dense space. Currently, the widely adopted methods learn the word embeddings from a large corpus, such as the *word2vec* model [1] using either *CBOW* or *Skip-gram* as the target objective, the *Glove* model [12] considering both the global statistics and local context, and *Elmo* [13] modeling words in a dynamic linguistic context.

Another kind of methods infer word embeddings based on their characters. Facebook AI Research [14] use n-gram as word features for English word representation learning. For Chinese, the *CWE* model [15] extracts the semantic information from Chinese characters and sums the embeddings of composed characters and subwords as a word embedding. Yu et al. [16] proposed the *JWE* model to jointly take into account the embeddings of words, subword and fine-grained character components. *JWE* predict the target word embedding by averaging the word embeddings, character embeddings and fine-grained character embeddings in the word context. Considering the special attributes of Chinese, morphological information are introduced to learn Chinese word embeddings. Yin et al. [17] proposed a Multi-Granularity Embedding (*MGE*) model, which represents the context as a combination of surrounding words, characters, and radicals of the target word. The *cw2vec* model [18] adopts the n-gram stroke features of a word instead of the word itself for training. For example, " 智 (wisdom)" is divided into " 知 (knowledge)" and " 曰 (say)". *GWE* [19] model can improve the effect by encoding from a bitmap through image convolution. Chen and Hu propose a dual channel network, where one is *CBOW* model and the other is character, component and radical information module [20]. Yang et al. [21] presented a character-enhanced Chinese word embeddings model (*CCWE*) based on *Skip-gram*. They introduced two tasks to train character and word embeddings simultaneously and the experimental results are better than the baseline models. These methods require a large corpus for training. If a word has less appeared in the training process, the word embedding can not be learned well.

To tackle the problem of rare appeared word embedding, Pinter et al. [7] used the embeddings of component characters and subwords. Experimental results showed that the performance of some advanced NLP tasks by these embeddings is better than with value <unk> on these words. Timo Schick et al. presented one-token approximation (OTA) [10], a method that obtains an embedding for a multi-token word and achieved great results in their own data set.After that, they introduce another *BERTRAM* [22], a model that integrates BERT into *Attentive Mimicking* [23]. In this way, they realized the deep integration of surface form and context, so as to obtain better representations for rare words. Hu et al. [9] proposed the attention-based hierarchical context model and adopt the cosine similarity as the objective function. This model does not encode the sequential usage of Chinese expression, which is often helpful for word semantics. Patel et al. [24] added subword co-occurrence information on the basis of word2vec. They combined the subword and context co-occurrence information linearly to learn the OOV word embeddings. These methods obtains a word embedding in an efficient way with only a few texts rather than a corpus. But they are not applicable for the specialty words since their semantics are less clarified in the usage text.

III. The Conceptual Text Embedding Model

In this paper, we consider the embedding problem with respect to conceptual definition or description for a Chinese new word. For a new word $w_u = c_1 c_2 \dots c_n$, we are given the concept description in the form of one or more pieces of text $t = w_1 w_2 \dots w_u \dots w_m$, where $c_i, i \in [1..n]$ is the word component character, and $w_j, j \in [1..m]$ is a word. The proposed model consists of two parts: the embedding generation and the semantic aggregation.

A. Embedding Generation.

We first generate a preliminary embedding by a BiLSTM using the characters of w_u, as illustrated at the top left in Fig.1. Comparing with marking the OOV word as a special value in traditional methods, this approach utilizes the characteristics of Chinese words since they often express general meanings or different aspects of the word. For example, 鲜花 (*Flower*) refers to fresh flowers. Each character, 鲜 (*Fresh*) or 花 (*Flower*), contains more general meanings. The final hidden states of bidirectional LSTMs are concatenated as the preliminary embedding $\boldsymbol{e_u}$ for word w_u, as equations 1-2.

$$\boldsymbol{h_i^c} = BiLSTM^c(\boldsymbol{c_i}, \boldsymbol{h_{i-1}}), i \in [1..n] \quad (1)$$

[1] github.com/Splab-Code/CTE

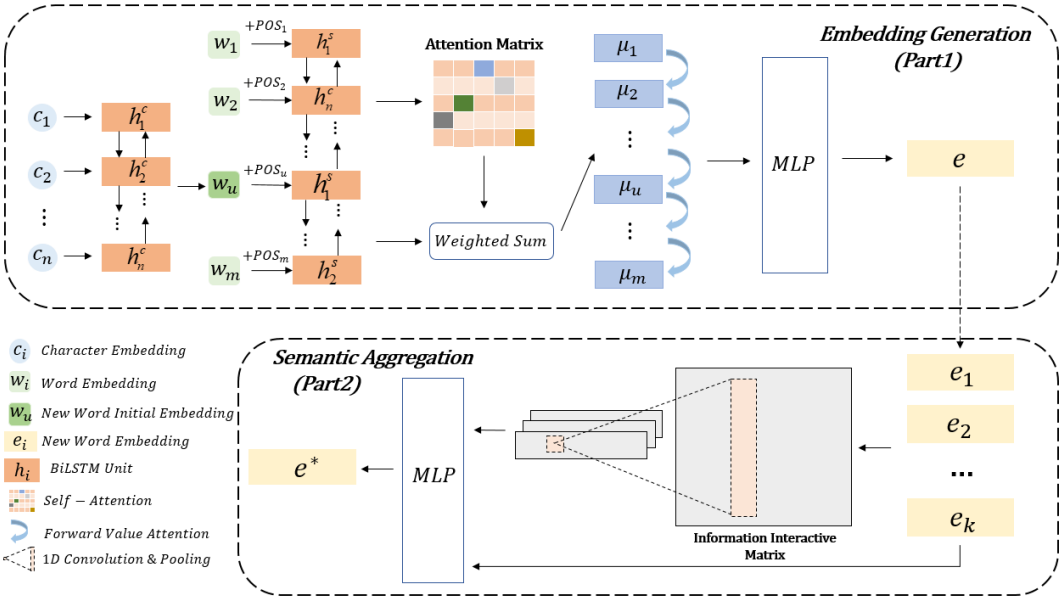


Fig. 1. The Conceptual Text Based Word Embedding Model.

$$\boldsymbol{e_u} = [\overrightarrow{\boldsymbol{h_n}}; \overleftarrow{\boldsymbol{h_n}}] \tag{2}$$

Then another BiLSTM is used to encode the sentence information, where each cell accepts the embedding $\boldsymbol{e_i}$ of word w_i and its part of speech information (POS) $\boldsymbol{e_{i\_POS}}$, as equations 3-4. The embeddings of POS categories are trained in the same process and the POS of w_u is set a specific value.

$$\boldsymbol{e_{w_i}} = \boldsymbol{e_i} \oplus \boldsymbol{e_{i\_POS}} \tag{3}$$

$$\boldsymbol{h_i^s} = BiLSTM^w(\boldsymbol{e_{w_i}}, \boldsymbol{h_{i-1}^s}), i \in [1..m] \tag{4}$$

The dual attention layers include the self-attention layer and the forward value attention layer. The purpose of the self-attention layer is to capture both the internal correlations and the global information of text, shown in equations 5-6. It accepts the hidden layer vector $\boldsymbol{h_i^s}$ and outputs $\boldsymbol{q_i}$ using the activation function. Each word gets a new feature vector $\boldsymbol{\mu_i}$ with the context information.

$$\boldsymbol{q_i} = \boldsymbol{\nu^T} tanh(W_\alpha \boldsymbol{h_i^s} + b_\alpha) \tag{5}$$

$$\alpha_i = \frac{exp(\boldsymbol{q_i})}{\sum_{t=1}^{m} exp(\boldsymbol{q_t})} \tag{6}$$

$$\boldsymbol{\mu_i} = \alpha_i \boldsymbol{h_i^s} \tag{7}$$

The second layer is the forward attention layer. Considering the sequential usage of Chinese expression, this forward attention can remain previous content for each step, as equations 8-9. Both the current step information $\boldsymbol{\mu_i}$ and the previous step vector $\boldsymbol{\mu_{i-1}}$ are combined together and $tanh$ is adopted for non-linear projection to $\boldsymbol{\omega}$.

$$\boldsymbol{f_i} = \boldsymbol{\omega}^T tanh(W_\delta \boldsymbol{\mu_{i-1}} + V \boldsymbol{\mu_i} + b_\delta) \tag{8}$$

$$\delta_i = \frac{exp(\boldsymbol{f_i})}{\sum_{t=1}^{m} exp(\boldsymbol{f_t})} \tag{9}$$

$$\boldsymbol{g} = \sum_{i=1}^{m} \delta_i \boldsymbol{\mu_i} \tag{10}$$

Finally, a MultiLayer Perceptron Network is used for embedding predication, i.e. $\hat{\boldsymbol{e}} = MLP(\boldsymbol{g})$. We adopt the Euclidean distance as the loss function to compare the predicted embedding with the pre-trained embedding $\boldsymbol{e}$, where λ is the regularization coefficient and $\boldsymbol{\theta}$ denotes the involved parameters. Since the residuals are very small, we take the form $log(x+1)$ in the implementation.

$$L = \|\hat{\boldsymbol{e}} - \boldsymbol{e}\|^2 + \lambda \|\boldsymbol{\theta}\|_2^2 \tag{11}$$

B. Semantic Aggregation

The aggregation model is proposed to combine the semantics of multiple texts, as the below part of Fig.1. Given $K \in N^+$ different texts, we can learn the set of embeddings $\{e_1, e_2, \cdots, e_K\}, e_k \in R^d, k \in [1..K]$ of these texts by the *Embedding Generation* model, where d is the dimension of embedding. For easy narration, we denote them by matrix $\boldsymbol{M} \in \mathbb{R}^{K \times d}$. We introduce two forms of interaction network for semantic aggregation. The 1-D CNN is applied for each column of $\boldsymbol{M}$ to encode the dimensional correlations and the MLP network encodes the interactions between dimensions.

We set L filters and let $W_l \in \mathbb{R}^{K \times 1}$ be the l-th filter weights. The filter W_l is applied to every column, where $\boldsymbol{m_j^T}$ denotes the operation on column j of matrix $\boldsymbol{M}$. Then the feature map $\boldsymbol{\gamma}$ is computed and the max pooling is performed on each dimension to get the final feature representation m'_j of column j, as equations 12-13. Finally, the K word embeddings $\{e_1, e_2, \cdots, e_K\}$ and the filter results $\boldsymbol{m} = [\mathring{m}_1; \mathring{m}_2; \cdots; \mathring{m}_d]$ are together fed into MLP for predicting the new word embedding, $\boldsymbol{e^*} = MLP([\boldsymbol{e_1}; \boldsymbol{e_2}; \ldots; \boldsymbol{e_K}; \boldsymbol{m}])$. We adopt the loss function similar with formula 11.

$$\boldsymbol{\gamma_{j,l}^T} = RELU(W_l \boldsymbol{m_j^T} + \boldsymbol{b_l}) \tag{12}$$

TABLE I
HYPER-PARAMETERS SETTING.

Hyper-parameter Name	Value
Char emb size	300
Word emb size	300
BiLSTM output dim	400
CNN num of filters	100
CNN filter size	[200.1]
Epochs	20
Learning rate	0.001
Dropout rate	0.5
Batch size	256

$$\mathring{m}_j = Max([\boldsymbol{\gamma}_{\boldsymbol{j,1}}^T, \boldsymbol{\gamma}_{\boldsymbol{j,2}}^T, \cdots, \boldsymbol{\gamma}_{\boldsymbol{j,L}}^T]) \tag{13}$$

IV. EXPERIMENTS

A. Data Set and System Settings

For the pre-trained Chinese word embeddings, we adopt the dataset provided by Li et al [25] since they have achieved good results in the word analogy experiments on both *CA-translated*[2] and *CA8 datasets*[3]. These embeddings were learned using the statistics on word co-occurrence and n-grams in Chinese Wikipedia corpus. Since it is difficult to find the set of words with conceptual texts, we first train our mode on this corpus by randomly choosing the texts where target words appeared and the occurrences of each word is larger than 5. The ratios on different types of word in the vocabulary are noun:verb:adjective:adverb 9:1:1:1. Then the model is fine tuned on a set of words with conceptual texts.

For test data, we select the words as the targets that neither appeared in the pre-training set nor are encountered during the model training. We also try different numbers of shots for each word to generate the embeddings so as to verify the performance of our aggregation model.

The models are trained on NVIDIA Geforce RTX 2080 Ti with Tensor-flow [26] and Adam optimizer [27]. The hyper-parameters used in our model are given in Table I. These parameters are obtained through many trials and we select the best.

B. Comparison Methods

- **Word2Vec(Mikolov et al.,2013)** is currently the most popular method that contains Skip-gram and CBOW models. Both are used as our baselines.
- **Glove(Pennington et al.,2014)** incorporates the information on both global matrix decomposition and local sliding window with a log-linear regression. It is also one of the most universal models at present.
- **JWE(Yu et al.,2017)** focuses on Chinese corpus, where the word embeddings are learned by adding radical information of Chinese characters. The work collected 13253 radicals in the experiments.

[2]github.com/Embedding/Chinese-Word-Vectors/tree/master/testsets/CA_translated

[3]github.com/Embedding/Chinese-Word-Vectors/tree/master/testsets/CA8

TABLE II
COMPARISON RESULTS ON WORD SIMILARITY TASK ($\rho \times 100$).

Methods	Word Similarity	
	Wordsim-240	Wordsim-296
Skip-gram*	44.2	44.4
CBOW*	47.0	50.2
Glove*	45.5	44.3
JWE*	48.0	52.7
Cw2vec*	**50.4**	52.7
EV-20	46.3	47.8
MIMIC17	47.2	49.5
CTE+1-shot	47.7	50.0
CTE+3-shot	49.9	53.2
CTE+5-shot	50.1	**53.4**

Baseline* results are from the paper (Cao et al. 2018)

- **Cw2vec(Cao et al.,2018)** exploits stroke-level information so as to encode the semantics and morphological information of Chineses.
- **MIMIC17(Pinter et al.,2017)** is an efficient way to learn the embedding of a new word by combining character information using RNN.
- **EV-20(Patel et al.,2020)** is a word2vec inspired model. It learns OOV word embeddings through the combination of the context clue and subword embeddings.
- **CTE** is our model.
- **CTEc** is a variant of our model. It adopts CNN instead of BiLSTM to process the character information of new word.

C. Word Similarity Task

The word similarity task directly tests the learnt embeddings on how well the semantic proximity and correlation between two words are modeled. We perform the task on data sets *wordsim-240* and *wordsim-296*[4], which are widely adopted to verify word embeddings. Each pair of words is manually given a score between 0-10 or 0-5 based on the relevance of the words. We compare the annotated scores with the similarity scores computed by different methods. The Spearman coefficient ρ is as an evaluation index of the dependence of two series of values. In the experiment, for each pair of words, we randomly select one to infer its embedding by using our model and the other is the pre-trained oracle embedding. To verify the influence of the number of texts for embedding aggregation, we adopt different shots, i.e. 1, 3 and 5 respectively, to learn word embeddings with our model.

From the experimental results in Table II, we can see that on dataset *wordsim-296*, our model achieves the best among all compared methods. Even with 3 shots, our model outperforms the best baseline result of *Cw2vec*, which illustrate the robustness of model on fewer texts. Besides, the performance of our method increases with an increasing number of texts. For dataset *wordsim-240*, our method is a little less than the best *Cw2vec*. The reasons reside on two sides: *Cw2vec* learned

[4]github.com/Leonard-Xu/CWE

TABLE III
PERFORMANCE COMPARISON ON DOWNSTREAM NLP TASKS.

Methods	TC	NER		POS Tagging	
	Acc	Boson-F1	MSRA-F1	Acc	New words Acc
Skip-gram	95.63	49.99	86.01	87.55	62.86
CBOW	96.97	50.21	86.82	87.52	62.17
Glove	97.05	51.86	86.77	87.63	63.04
MIMIC17	97.66	57.93	88.42	88.35	70.65
CTEc	97.82	59.02	89.23	89.22	72.31
CTE	**98.19**	**61.17**	**89.89**	**90.36**	**75.22**
Our Model with Different Components					
COWE.Part1 w/o Character	97.70	57.92	88.49	88.47	63.11
COWE.Part1 w/o POS	98.11	60.88	89.29	90.01	64.68
COWE.Part1 w/o Self-attention	97.69	57.97	88.51	88.20	62.89
COWE.Part1 w/o Forward-attention	97.78	58.93	89.02	89.14	63.13
COWE.Part2 w/o CNN	97.74	58.52	88.96	88.49	62.96

these embeddings on a large corpus and the words in *wordsim-240* are very popular with high frequencies in the corpus. Even for this case, we still get a comparatively high score 50.1 with only a few sentences.

Comparing with the methods that are designed for solving the new word embedding problem MIMIC17 and EV-20, MIMIC17 achieves a higher score. EV-20 method uses sub-word embedding information by new word internal two-gram. For example, the word 'clue' are divided into '<c','cl','lu','ue' and 'e>'. However, two-character words are the most common in Chinese words. The EV-20 method does not benefit from the above operation on Chinese. Therefore, we choose MIMIC17 as the main comparison method for subsequent experiments.

D. Evaluation on Downstream Tasks

a) Text Classification(TC): TC is often chosen as the natural language downstream task for verifying word embeddings. We use the Fudan Chinese classification corpus[5]. We select five categories with a large number of documents: environment, agriculture, finance, politics and sports. To mimic new words, we randomly erase 10% words of the vocabulary in this dataset. For the text classification task, we employ the text BiLSTM method and use accuracy as the measure.

b) Named Entity Recognition(NER): NER is a semantic task that identifies the entities with specific meanings in text. In this experiment, we use BiLSTM-CRF [28] method to perform experiments on two datasets: BosonNLP[6] and MSRA[7]. There are 1350 new words among BosonNLP. These new words account for 11.61% in the total vocabulary. BosonNLP contains seven entity types, which include time, place name, person name, company name, organization name, product name and etc. We randomly select 70% of data for training and 30% for text. MSRA is provided by Microsoft Research Asia. There are three types of named entities: person names, place names, and organization names, and with 2084 new words. We process the data into word-level forms and label them in BIO format. We use F1-score as an evaluation metric to compare different methods.

c) Part-of-speech tagging(POS Tagging): This task marks the part-of-speech of words in a sentence according to their meanings and context. We adopt the 2014 People's Daily Corpus[8] as the experimental data, which contain 52,847 articles and 15,890 new words. We use the RNN classification algorithm for this task. The performance is measured by accuracy.

Results. We present the comparison results on these NLP Downstream tasks in Table III, where the methods *MIMIC17* and *CTEc* are specially designed for the new word embedding problem. We can see that the lack of new word embedding has an impact on the accuracy of the above tasks. In the text classification experiment, our model obtains the best result of 98.19. Compared with the best baseline model GloVe, the accuracy improved by 1.14%, and compared with the MIMIC17 model that is specifically designed for new word embeddings, our score is also increased by 0.53%. As for the NER task, our model performs the best on both datasets. Since the major proportion of new words in BosonNLP is much larger than that of the MSRA dataset, i.e. 1026 out of 1350 vs 386 out of 2084, the improvement on BosonNLP is larger than on MSRA. For the part-of-speech tagging task, our method also outperforms baseline methods, which achieves the highest scores of 90.36 and 75.22. The accuracy of our method is larger than MIMIC17 by nearly 5% improvement.

E. Model Validation

This section examines how much the components of our model affect the performance by the tasks in Sec. IV-D. The results are reported in the Table III, which show that every component positively contributes to the performance. The self-attention layer contributes the most, where the differences are

[5]download.csdn.net/download

[6]bosonnlp.com/dev/resource

[7]download.csdn.net/download/shuihupo

[8]download.csdn.net/download/10270189

TABLE IV
EMBEDDING BASED RANKING EXAMPLES ON DIFFERENT NUMBER AND TYPES OF TEXTS FOR WORD COMPARISON

Methods	Shot#	Top-5 similar words(cosine similarity)
n-CTE	1	经济报 (*Economic news*), 经济网 (*Economic net*), 评论部 (*Comment department*), 中广网 (*CATV net*), 科技网 (*Stdaily net*)
	3	经济网 (*Economic net*), 京华网 (*Jinghua net*), 中广网 (*CATV net*), 人事网 (*Cpta-net*), 俄通社 (*ITAR*)
	5	中广网 (*CATV net*), 经济网 (*Economic news*), 人大网 (*People's Congress*), 红网 (*Red net*), 论坛网 (*Forum Net*)
l-CTE	1	人民日报 (*People's daily*), 千龙网 (*Qianlong*), 新华网 (*Xinhua net*), 本报 (*Newspaper*), 工人日报 (*Worker's daily*)
	3	人民日报 (*People's daily*), 经济网 (*Economic net*), 法制日报 (*Legal Daily*), 新华网 (*Xinhua net*), 千龙网 (*Qianlong*)
	5	新华网 (*Xinhua net*), 法制网 (*Legal net*), 经济网 (*Economic net*), 人民日报 (*People's daily*), 千龙网 (*Qianlong*)
CTE	1	人民日报 (*People's daily*), 千龙网 (*Qianlong*), 新华网 (*Xinhua net*), 本报 (*This Newspaper*), 工人日报 (*Worker's daily*)
	3	新华网 (*Xinhua net*), 论坛网 (*Forum Net*), 经济网 (*Economic net*), 人大网 (*People's Congress*), 人民日报 (*People's daily*)
	5	新华网 (*Xinhua net*), 青年网 (*Youth net*), 中广网 (*Catv net*), 科技网 (*Stdaily-net*), 经济网 (*Economic net*)
GOLD	-	新华网 (*Xinhua net*), 中广网 (*Catv net*), 中新网 (*Ecns net*), 正义网 (*Justice-net*), 经济网 (*Economic net*)

0.5%-3.25% on several tasks comparing with the best results of our model. Without character component or the forward-attention layer, the accuracy drops by 0.49%-3.2% and 0.41%-2.24%, respectively. The introduce of CNN in the aggregation model also brings an improvement of 0.35%-2.65%.

F. Evaluation on The Number and Types of Texts

This task evaluates whether the learned word embeddings are compatible with pre-trained word embeddings. We use different shots and types of conceptual texts to learn the embedding of a word and compute the similarity between the target word and pre-trained word. The golden result is calculated using the pre-trained embedding of the target words.

We choose different types of text as the input of our model for comparison, conceptual text and non-conceptual text, denoted by **CTE** and **n-CTE**. Taking the Chinese word 人民网 (*People's Daily Online*) as an example, the conceptual text is 人民网是世界十大报纸之一《人民日报》建设的以新闻为主的大型网上信息交互平台 (*People's Daily Online is built a large-scale online information interaction platform based on news by one of the world's top ten newspapers ⟨the People's Daily⟩*). A non-conceptual text refers to a text where the target appears as a usage example, such as 人民网报道，北京座谈会现场为充分挖掘市场监管领域工作亮点 (*People's Daily Online reported that on the Beijing Symposium site, people fully explored the highlights of market supervision...*). As the comparison with the aggregation model, we select the linear combination of multiple embeddings, denoted by **l-CTE**. We also quantify how the number of texts influence the quality of word embeddings.

We list the top-5 similar words with *People's Daily Online* in Table IV. The results show that more texts result in better word embeddings. Using the same conceptual texts for aggregation, our method is better than **l-CTE**. For example, on 3-shot aggregation, the word *Xinhua net* is at the first position by our aggregation method.

In addition to the word 人民网 (*People's Daily Online*), we also choose other Chinese words to illustrate the effectiveness of our model. Due to the limitation of paper space, we use the ranking indicators $NDCG@5$ and $NDCG@10$ to evaluate the results, as listed in Table V. The results are the average values of randomly sampling from candidate texts for 10 times. As shown in Table V, comparing conceptual texts and non-conceptual texts, the values of NDCG@5 and NDCG@10 increase with the increasing number of sentences. And the word embedding trained with conceptual texts is more semantic than those trained with non-conceptual texts. Overall, our method achieved the highest scores on different metrics, which show that the aggregation model outperforms other methods.

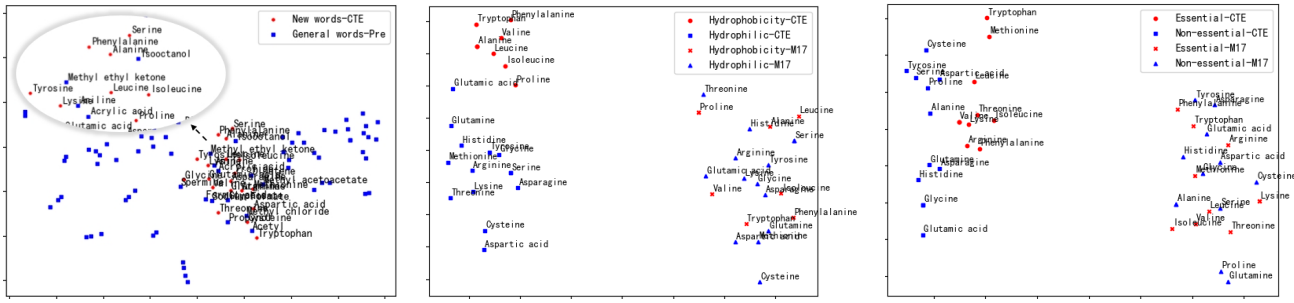
G. Analysis on Specialty Word Embeddings

We choose the specialty domain Biology and verify the embedding results for 20 amino acids. Their conceptual texts are chosen from the Chinese encyclopedia[9]. In this experiment, 5 conceptual texts are used for each specialty concept noun for computing its embedding. In Fig. 2, we use the T-SNE [29] method to visualize the embeddings of general words and new words.

[9]baike.baidu.com

TABLE V
QUANTITATIVE EVALUATION ON NEW WORD EMBEDDING.

Methods	Shot#	人民网 (*People's Daily Online*)		备付金 (*Provision*)		唢呐 (*Suona*)	
		NDCG@5(%)	NDCG@10(%)	NDCG@5(%)	NDCG@10(%)	NDCG@5(%)	NDCG@10(%)
n-CTE	1	22.86	25.19	11.18	22.63	14.91	20.52
	3	27.69	38.77	21.32	28.36	32.58	30.97
	5	43.66	50.65	48.22	49.49	43.94	49.65
	7	45.32	59.78	51.98	52.68	46.01	52.79
l-CTE	1	24.31	30.26	19.23	25.43	20.31	28.17
	3	26.71	55.74	45.89	50.86	34.55	42.34
	5	50.22	63.31	56.24	60.49	51.62	56.47
	7	55.19	67.16	58.62	61.87	52.66	58.21
CTE	1	24.31	30.26	19.23	25.43	20.31	28.17
	3	50.24	56.45	51.65	51.26	45.88	50.38
	5	64.98	68.97	62.74	65.46	52.67	59.77
	7	**69.95**	**71.28**	**65.67**	**69.26**	**55.01**	**61.88**



(a) Semantic embedding distribution. (b) Using the structural properties texts. (c) Using the functional properties texts.

Fig. 2. T-SNE visualization on 20 Amino Acids Embeddings.

The relationship between the word embeddings learned by our model and the pre-trained word embeddings are shown in Fig. 2(a). We can observe that the 20 amino acids words cluster closely than with other general words. Besides, there are many semantically similar words around them. For example, both the words *Aniline* and *Acrylic acid* are related to the composition of *Lysine*. So they are very close to *Lysine*. These illustrate that the similar attributes they have are learned by our model.

We then analyze whether our model captures the differences of text. We select two specialty attributes in the description texts, i.e. *space structure* and *bio-function*, and present the results in Fig. 2(b) and Fig. 2(c), where *MIMIC17* is as the comparison method. Here are the texts about the two properties of 甘氨酸 (*Glycine*): 甘氨酸在水中易溶，在分子中同时具有酸性和碱性官能团，具有很强的亲水性 (*Glycine is easily soluble in water. Its molecule has both acidic and basic functional groups so it has strong hydrophilicity*) and 甘氨酸不一定非从食物直接摄取，属于非必须氨基酸，可以在自身内合成 (*Glycine is not necessarily ingested directly from food. It is a non-essential amino acid and can be synthesized in itself*). The left side in each Fig. 2(b) and Fig. 2(c) uses the embeddings obtained by our model and the right side is the result of *MIMIC17*. The results show that amino acids with the same attribute appear closer than that by the baseline method. For example, 甘氨酸 (*Glycine*), 丝氨酸 (*Serine*) and 赖氨酸 (*Lysine*) in Fig. 2(b) are all hydrophilic. In contrast, 甘氨酸 (*Glycine*) is farther away from 亮氨酸 (*Leucine*) with hydrophobic properties. In Fig. 2(c), the amino acids presents two clusters because of the functional differences. These results show that our model outperforms *MIMIC17* on learning the word semantics from the conceptual texts.

V. CONCLUSION

In this paper, we present a two-stage model to learn a new word embedding by the conceptual text, where the first one encodes the information of word components and the context of description, and the second aggregates the semantics of multiple embeddings. We perform experiments on six datasets to verify the proposed method and the results outperform the state of art methods on both direct semantics verification and advanced NLP tasks. We also experimentally verify the effects of different parts of model , the number and types of conceptual texts. Finally, we present some biology texts to illustrate

whether the embeddings have encoded the semantics of new words in the specialty. In future, we will consider extensions in pragmatics. We would try the effects of differential expressions of conceptual texts and make the results more suitable for professional fields.

Acknowledgment

This work was supported by the Major Project of NSF Shandong Province (ZR2018ZB0420), the National Key R&D Program of China (2018YFC0831401,2018YFC0831406), the National Natural Science Foundation of China (91646119), and the Key Research and Development Project of Shandong Province(2019JZZY010107).

References

[1] T. Mikolov and K. C. et al, "Efficient estimation of word representations in vector space," in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013*, 2013.

[2] E. Grave and T. M. et al, "Bag of tricks for efficient text classification," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017*, 2017, pp. 427–431.

[3] Y. Zhang and J. Yang, "Chinese NER using lattice LSTM," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018*, 2018, pp. 1554–1564.

[4] A. Vaswani, N. Shazeer, and N. P. et al, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, 2017, pp. 5998–6008.

[5] D. Chaudhuri and A. K. et al, "Improving response selection in multi-turn dialogue systems by incorporating domain knowledge," in *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, 2018, pp. 497–507.

[6] I. Bazzi, "Modelling out-of-vocabulary words for robust speech recognition," Ph.D. dissertation, Massachusetts Institute of Technology,MIT, 2002.

[7] Y. Pinter, R. Guthrie, and J. Eisenstein, "Mimicking word embeddings using subword rnns," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, 2017, pp. 102–112.

[8] Y. Zhang and Y. e. a. Liu, "Learning chinese word embeddings from stroke, structure and pinyin of characters," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. ACM, 2019, pp. 1011–1020.

[9] Z. Hu and T. C. et al, "Few-shot representation learning for out-of-vocabulary words," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019*, 2019, pp. 4102–4112.

[10] T. Schick and H. Schütze, "Rare words: A major problem for contextualized embeddings and how to fix it by attentive mimicking," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA*. AAAI Press, 2020, pp. 8766–8774. [Online]. Available: https://aaai.org/ojs/index.php/AAAI/article/view/6403

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[12] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[13] M. E. Peters, M. Neumann, and M. I. et al, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018*, 2018, pp. 2227–2237.

[14] P. Bojanowski and E. e. a. Grave, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[15] X. Chen, L. Xu, and Z. L. et al, "Joint learning of character and word embeddings," in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, 2015, pp. 1236–1242.

[16] J. Yu and X. J. et al, "Joint embeddings of chinese words, characters, and fine-grained subcharacter components," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, 2017, pp. 286–291.

[17] R. Yin and Q. e. a. Wang, "Multi-granularity chinese word embedding," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, 2016, pp. 981–986.

[18] S. Cao and W. e. a. Lu, "cw2vec: Learning chinese word embeddings with stroke n-gram information," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[19] T.-r. Su and H.-y. Lee, "Learning chinese word representations from glyphs of characters," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, September 2017, pp. 264–273.

[20] Z. Chen and K. Hu, "Radical enhanced chinese word embedding," in *Chinese computational linguistics and natural language processing based on naturally annotated big data*. Springer, 2018, pp. 3–11.

[21] G. Yang, H. Xu, T. He, and Z. Cai, "A character-enhanced chinese word embedding model," in *International Joint Conference on Neural Networks, IJCNN 2019 Budapest, Hungary, July 14-19, 2019*, 2019, pp. 1–5.

[22] T. Schick and H. Schütze, "BERTRAM: improved word embeddings have big impact on contextualized model performance," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, Eds. Association for Computational Linguistics, 2020, pp. 3996–4007. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.368/

[23] ——, "Attentive mimicking: Better word embeddings by attending to informative contexts," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019*, J. Burstein, C. Doran, and T. Solorio, Eds. Association for Computational Linguistics, 2019, pp. 489–494. [Online]. Available: https://doi.org/10.18653/v1/n19-1048

[24] R. Patel and C. Domeniconi, "Estimator vectors: OOV word embeddings based on subword and context clue estimates," in *2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020*, 2020, pp. 1–8.

[25] S. Li and Z. Z. et al, "Analogical reasoning on chinese morphological and semantic relations," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018*, 2018, pp. 138–143.

[26] M. Abadi and X. Z. et al, "Tensorflow: A system for large-scale machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016, Savannah, GA, USA, November 2-4, 2016*, K. Keeton and T. Roscoe, Eds. USENIX Association, 2016, pp. 265–283. [Online]. Available: https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[28] X. Ma and E. H. Hovy, "End-to-end sequence labeling via bi-directional lstm-cnns-crf," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany*, 2016.

[29] J. Hunter, "Tsnet - A distributed architecture for time series analysis," in *Computer-based Medical Guidelines and Protocols: A Primer and Current Trends*. IOS Press, 2008, vol. 139, pp. 223–232.