Enriching Chinese Word Embeddings with Characters and Fine-grained Subcharacter Components

Jinxing Yu **Xun Jian** **Hao Xin** **Ke Zhang**

Abstract

Different from most western language, Chinese is a kind of hieroglyphs and contains rich morphological information. In this work , we propose an approach to enriching chinese word embeddinga with characters and finegrained subcharacter components information. Evaluation on both word similarity and word analogy task demonstrates the superior performance of our model.

1 introduction

Distributed word representation embeds a word into a continuous low dimentional vector and can better uncover both the semantic and syntactic information over traditional bag of words representations. It has been successfully applied in many downstream NLP tasks as input features, such as named entity recognition, sentiment analysis, and question answering. Among many embedding methods, CBOW and Skip-Gram model are very popular due to their simplicity and efficiency, making it feasible to learn good embeddings from a large scale training corpus.

Despite the success and popularity of word embeddings, most of the existing methods treat each word as the mininum unit, which ignores the morphological inormation of words. The representation of rare words may be poor despite that the training process of CBOW and Skip Gram typically subsamples the frequent words. To address this issue, many recent works have investigated how to leverage morphological inormation to learn better word embeddings. It has been proved efficive to improve word embedding quality, especially for morphologically rich languages.

Chinese is a kind of hieroglyphs and is a morphologically rich language. The characters compsing a word can indicate the semantic meaning of the word and words sharing same character components always have similar meanings. Moreover, chinese characters can be broken into fine-grained components, which can be roughly divided into two types: semantic component and phonetic component. The semantic component represents the meaning of a character while the phonetic component represents the sound of a character.

Leveraging these subword informations such as characters and character components can enrich chinese word embeddings with internal morphological semantics. Some methods have been proposed to incorporate these subword information for chinese word embeddings. Sun Y et al. 2014 first introduced a radical-enhaced chinese character embedding model based on C&W model and and apply it on Chinese character similarity judgement and Chinese word segmentation. Yanran Li 2015 et al. developed two component-enhanced Chinese character embedding models and their bigram extensions based on CBOW and Skip Gram model. Chen et al. proposed CWE model to joint learn chinsed word and character embedding and utilize the chinese characters to enrich chinese word embeddings. Jian Xu et al 2016 extends CWE work by exploiting the internal sematic similarity between a word and its characters and combining word and character embedding in a cross-lingual fashion. To combine the radical-character, character-word composition information, Rongchao Yin et al. 2016 propose multi-granularity embedding (MGE) model based on CWE model, which repsents the context as the combination of surrounding words, surrounding characters and the radical of target word to predict the target word.

However, previous works only use character or radical of character to enrich a word embedding and don't make full use of fined grained components of characters. The component of chinese character is different from radicals, they are sometimes wrongly considered the same. Essentially,

radicals are a specific set of characters that are used to index Chinese characters in dictionaries. Although many of them (not all) are also semantic components, each character has only one radical, which can not fully uncover the semantic and structure of a character. Besides, there are about 200 radicals while the number of components is over 10000. Xinlei Shi et al. 2016 cut a character into fine-grained components according to wubi input method and get a these component embeddings by CBOW model. Then it feeds these component embeddings into deep neural networks and achives promising results in short-text categorization, chinese word segmentation, web search ranking tasks.

In this work, we present a model to jointly learn chinese word, character, sub character components embeddings. The learned chinese word embeddings can not only leverage the external context concurrence information but also incorporate rich internal subword structure and semantic information. Expreiments on both word similarity and word analogy tasks demonstrate the effectiveness of our model over previous works.

2 Joint Learning Model

In this section, we introduces our joint leraning model, which combine word, character, sub character component information. Our model is based on CBOW model, we compare the effectiveness of two sub character features: radical, component and two methods that combines word, character, sub character vectors in the context: JOIN1 and JOIN2. JOIN1 borrows the idea of BEING (Fei Sun et al. 2016) and uses the sum of word vectors, the sum of character vectors, the sum of sub characters to predict te target word seperately and sum these three predict loss as the final loss function. JOIN2 follows the idea of CWE(Chen et al.) and represents the word in the context as the composition of word embeddings , its characters embeddings and its sub character embeddings. Observing that in multi-granularity embedding (MGE) model(Yin et al. 2016), the radicals of the target word are used in the context to predict the target word, we also compare the performance of combining the surrounding words'sub character information and combining the target word's sub character information in our model.

Let $D = (w_1, w_2, \cdots, w_n)$ be the training corpus, $C = (c_1, c_2, \cdot, c_m)$ be the set of characters, $S = (s1, s_2, \cdots, s_l)$ be the set of sub characters, K be the window size.

JOIN1 For JOIN1 combining method, we aim to maximize the sum of three predictive loss for a target word w_i:

$$L(w_i) = \log P(w_i|h_{i1}) + \log P(w_i|h_{i2}) + \log P(w_i|h_{i3})$$

where h_{i1}, h_{i2}, h_{i3} is the composition of context word embeddings, context character embeddings, context sub character embeddings resprectively. More precisely, we denote the surrounding words, characters, sub characters as the context , they can be represented as following:

$$h_{i1} = \sum_{w_j \in context} w_j$$

$$h_{i2} = \sum_{c_j \in context} c_j$$

$$h_{i3} = \sum_{s_k \in context} s_k$$

The conditionaly probability is defined by the soft max function

$$p(w_i|h_{i_k}) = \frac{\exp(h_i^T w_{i_k})}{\sum_{j=1}^{n} \exp(h_{i_k}^T w_j)} \text{for k = 1, 2, 3}$$

The model aims to maximize the overall log likelihood :

$$L(D) = \sum_{i=1}^{n} L(w_i)$$

JOIN2 For JOIN2 combining method, the training objective is to maximize the following overall log likelihood:

$$L(D) = \sum_{w_i \in D} \log p(w_i|h_i)$$

where h_i is the vector composed by the embedding of context words, characters, and sub characters.

$$h_i = \sum_{t=i-K, t\neq i}^{i+K} \frac{1}{2K} (w_t + \sum_{c_j \in w_t} \frac{1}{|w_t|} (c_j + \sum_{s_k \in c_j} \frac{1}{|c_j|} s_k))$$

$|w_i|$ represents the number of characters in word w_i and $|c_j|$ represents the number of sub characters in charater c_j.

The conditional probability is defined as

$$p(w_i|h_i) = \frac{\exp(h_i^T w_i)}{\sum_{j=1}^{n} \exp(h_i^T w_j)}$$

3 Experiment Setup

Training Data We use the Chinese Wikipedia as our training data source. In Chinese sentences, words are not separated by special symbols (as space in English sentences), so we firstly use THULAC[1] as the lexical analysis tool to separate words in sentences, then we can use this formated corpus in our experiments.

Character Components We crawled the component and radical information of Chinese characters from HTTPCN[2]. This dataset contains 20879 characters, 13253 components and 218 radicals, of which 7744 characters have more than one components, and 214 characters are equal to their radicals.

Parameter Settings We fix the word vector length to be 200, the window size to be 5, and the training iteration to be 1. Words with frequency less than 5 are ignored because they are too rare. The negative sampling size is set to be 10 and the subsampling parameter is set to be 10^{-3}.

Baseline We use the CBOW model in work **CWE** as the baseline model. All the parameters are set to be the same as those in our model.

Similarity Evaluation Metrics In this part, we evaluate the quality of an embedding by a ranking-correlation method. For all 3-tuples (w_1, w_2, s) in a similarity testing dataset, we can calculate the similarity s' between w_1 and w_2 with an embedding, then we calculate the Spearsman Correlation between all the s and s' as the quality of this embedding.

Analogy Evaluation Metrics In the analogy testing dataset, let (w_1, w_2, w_3, w_4) be a tuple, then with a 'good' word embedding e_i of each word w_i, we can write down this form

$$\begin{aligned} & e_2 - e_1 \approx e_4 - e_3 \\ \Rightarrow & e_2 - e_1 + e_3 \approx e_4 \\ \Rightarrow & (e_2 - e_1 + e_3) \cdot e^{(\ell)} \approx e_4 \cdot e^{(\ell)}. \end{aligned}$$

This form shows $(e_2 - e_1 + e_3) \cdot e^{(\ell)}$ is an approximation of $e_4 \cdot e^{(\ell)}$, which is the similarity between e_4 and $e^{(\ell)}$. If we calculate $(e_2 - e_1 + e_3) \cdot e^{(\ell)}$ for each $w^{(\ell)}$, we are expected to get the maximum result when $w^{(\ell)} = w_4$. In our experiments, with a word embedding and a tuple, we pick the word with maximum approximate similarity (except for the first three words in this tuple) as the prediction of the fourth word, then the prediction precision over all tuples are used as the measurement of the quality of this embedding.

[1] http://thulac.thunlp.org/

[2] http://tool.httpcn.com/

4 Results

4.1 Human similarity judgement

In this task, we use two different chinese similarity datasets, wordsim-240 and wordsim-296, which are proposed py Xinxiong Chen e tal. In wordsim-240, there are 240 pairs of Chinese words and human-labeled relatedness scores, of which the 233 word pairs have appeared in the learning corpus. In wordsim-297, the words in 280 word pairs have appeared in the learning corpus and the left 16 pairs have new words. We compute the Spearman correlation between relatedness scores from a model and the human judgements for comparison. The evaluation results of our model and baseline methods on wordsim-240 and wordsim-296 are shown in Table 1 From the results, we observe that: (1) Our models with jion type 1 compute the semantic relatedness of these word pairs much closer to human judgements. (2) The models use information of componant outperform the models using radicals. The reason is that, these components may contains more semantic information about the character. (3) The information of the character and components/radicals of the target word is more useful than those of the context. This can be explained that the influnce of the components mainly stays in the character level, will not influence the meaning of the surrounding words. The change of datasets does not cause significant change of correlations for both baselines and our methods.Our models compute the semantic relatedness of these word pairs much closer to human judgements.

model	wordsim-240	wordsim-297
cbow	50.88	61.87
c-comp-j1[1]	55.44	57.70
c-comp-j2	21.28	34.57
c-radi-j1	54.37	64.75
c-radi-j2	23.82	34.40
t-radi-j1	54.70	63.09
t-comp-j1	55.62	65.61
t-comp-j2	21.40	27.58
t-radi-j2	21.77	27.84

Table 1: Similarity results

4.2 Analogical Reasoning

In this task, we use the Chinese dataset introduced by Xinxiong Chen e tal. which consists of 1124 tuples of words and each tuple contains 4 words, coming from three different categories 'Capital', 'State' and 'Family'. The words w_i in each tuple (w_1, w_2, w_3, w_4) in this dataset have the relationship that w_2 is to w_1 as w_4 is to w_3. The learning corpus covers more than 97% of all the testing words. From Table 2, we observe that: (1) Our

model	Total	Capital	State	Family
cbow	61.29	66.91	73.71	39.33
c-comp-j1[1]	41.28	42.84	53.14	29.78
c-comp-j2	17.08	17.13	22.86	13.24
c-radi-j1	68.32	72.71	88.00	47.43
c-radi-j2	16.90	16.94	22.29	13.97
t-radi-j1	67.62	70.61	90.86	45.22
t-comp-j1	68.50	73.70	91.42	40.80
t-comp-j2	17.08	17.13	22.86	13.24
t-radi-j2	16.90	16.94	22.29	13.97

Table 2: Analogical results

models with jion type 1 and components information of the character in target word significantly outperform baseline methods. This indicates the necessity of considering character and components embeddings for word embeddings. However, it's also very important to consider the way to combine the information (2) This models can improve the embedding quality of all words, not only those words whose characters are considered for learning. For example, in the type of capitals of countries, all the words are entity names whose characters are not used for learning. Our model can still make an improvement on this type as compared to baseline models. (3) However, we also observe that the component of characters in surounding words may confuse the model, and lead to a bad result. And we think this is highly related to the origin construction of chinese characters and chinese words, that is components can have only local influnce, while characters may influnce the meaning of the surrounding words.

5 Conclution

In this work, we propsed a model to jointly learn chinese word, character, sub character embeddings. Our approach make full use of characters and finegrained sub characters information to enrich chinese word embeddings. Experiments show the benefit to incorporating finegrained components compared to just using radicals or characters.

It seems that we have some errors in the implementation of JOIN2 method, we will fix it in the future. We also find that we may have a slit problem in the preprocessing of chinese wikipedia dump since the performance of CBOW in our experiment is slitly differment from that in a previous work that uses chinese wikipedia dump(And itt use windowsize 3 while CWE use windowsize 5 and people daily news as training corpus). We haven't made a good comparasion with all previous works. We will finish the experiment in the future.

[1] c=context,t=target,comp=component,rad=radical