Generating Abbreviations for Chinese Named Entities Using Recurrent Neural Network with Dynamic Dictionary

Qi Zhang, Jin Qian, Ya Guo, Yaqian Zhou, Xuanjing Huang
Shanghai Key Laboratory of Data Science
School of Computer Science, Fudan University
Shanghai, P.R. China
{qz, jqian12, yguo13, zhouyaqian, xjhuang}@fudan.edu.cn

Abstract

Chinese named entities occur frequently in formal and informal environments. Various approaches have been formalized the problem as a sequence labelling task and utilize a character-based methodology, in which character is treated as the basic classification unit. One of the main drawbacks of these methods is that some of the generated abbreviations may not follow the conventional wisdom of Chinese. To address this problem, we propose a novel neural network architecture to perform task. It combines recurrent neural network (RNN) with an architecture determining whether a given sequence of characters can be a word or not. For demonstrating the effectiveness of the proposed method, we evaluate it on Chinese named entity generation and opinion target extraction tasks. Experimental results show that the proposed method can achieve better performance than state-of-the-art methods.

1 Introduction

Abbreviations of Chinese named entities are frequently used on different kinds of environments. Along with the development of social media, this kinds of circumstance occurs more frequently. Unlike western languages such as English, Chinese does not insert spaces between words or word forms that undergo morphological alternations. Hence, most of the Chinese natural language processing methods assume a Chinese word segmenter is used in a pre-processing step to produce word-segmented Chinese sentences as input. However, if the Chinese word segmenter produces erroneous output, the quality of these methods will be degraded as a direct result. Moreover, since the word segmenter may split the targets into two individual words, many methods adopted character-based methodologies, such as methods for named entity recognition (Wu et al., 2005), aspect-based opinion mining (Xu et al., 2014), and so on.

Through character-based methodology, most of the previous abbreviation generation approaches have been formalized as sequence labelling problem. Chinese characters are treated as the basic classification unit and are classified one by one. In these methods, dictionaries play important effect in constructing features and avoiding meaningless outputs. Various previous works have demonstrated the significant positive effectiveness of the external dictionary (Zhang et al., 2010). However, because these external dictionaries are usually static and preconstructed, one of the main drawbacks of these methods is that the words which are not included in the dictionaries cannot be well processed. This issue has also been mentioned by numerous previous works (Peng et al., 2004; Liu et al., 2012).

Hence, understanding how Chinese words are constructed can benefit a variety of Chinese NLP tasks to avoid meaningless output. For example, to generate the abbreviation for a named entity, we can use a binary classifier to determine whether a character should be removed or retained. Both “国航” and “中国国航” are appropriate abbreviations for “中国国际航空公司(Air China)”. However “国航司” is not a Chinese word and cannot be understood by humans.

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 721–730,
Austin, Texas, November 1-5, 2016. ©2016 Association for Computational Linguistics

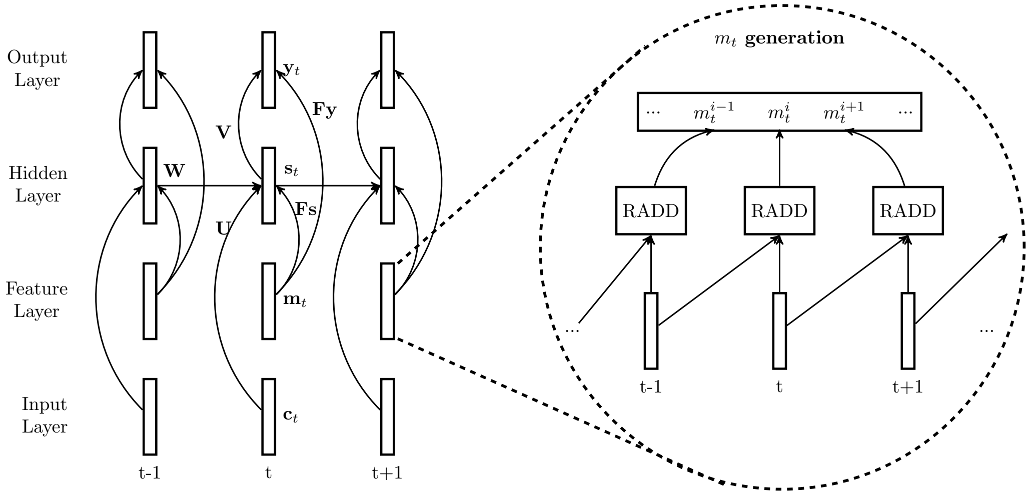

Figure 1: The architecture of RNN with dynamic dictionary.

Thus we are motivated to study the task of "*dynamic dictionary*" and integrating it with sequence labelling model to perform the abbreviation generation task. Dynamic dictionary denotes a binary classification problem which tries to determine whether or not a given sequence of characters is a word. Although human can use implicit knowledge to easily recognize whether an unseen text segment is a word or not at first glance, the task is not as easy as it may seem. First, Chinese has a different morphological system from English. Each Chinese character represents both a syllable and a morpheme (McBride-Chang et al., 2003). Hence, Chinese script is sometimes described as being morphosyllabic. Second, there are many homophones in Chinese. This means that characters that have very different written forms may sound identical. Third, there are a huge number of Chinese words. Without taking the implicit knowledge of morphology into consideration, an arbitrary sequence of characters can be used as a name. In Mandarin, there are approximately 7,000 characters in daily use. Hence, determining whether a given sequence of characters is a word or not is an challenging task.

Since the length of Chinese words is variable, in this paper, we propose a modified recurrent architecture to model the dynamic dictionary construction task. For processing sequence labelling tasks, we also combine the proposed method with RNN. Since the proposed dynamic dictionary model can be pre-trained independently with extensive domain independent dictionaries, the combined model can be easily used in different domains. The proposed model can take advantage of both the sequence-level discrimination ability of RNN and the ability of external dictionary.

The main contributions of this work can be summarized as follows:

- We define the dynamic dictionary problem and construct a large dataset, which consists of more than 20 million words for training and evaluation.

- We integrate RNN with a deep feedforward network based dynamic dictionary learning method for processing Chinese NLP tasks which are formalized as sequence labelling tasks.

- Experimental results demonstrate that the accuracy of the proposed method can achieve better results than current state-of-the-arts methods on two different tasks.

2 Model Architecture

2.1 Dynamic Dictionary

The task of dynamic dictionary is to predict whether a given sequence of characters can be a word or not. The input is a text segment, which contains a variable number of characters. The output is an

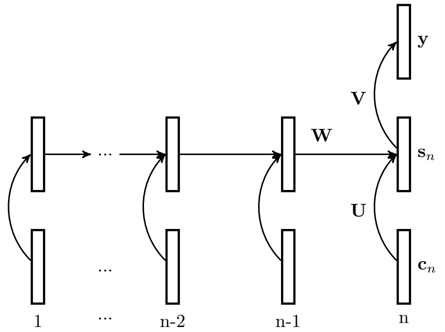

Figure 2: The recurrent architecture used in this work for modelling dynamic dictionary (RADD).

binary value. It is different from the traditional sequence classification tasks, whose the number of outputs are usually same as the input. However, the information of the whole sequence is an important factor and should be incorporated. Hence, in this work, we use a modified recurrent architecture (RADD is used to represent the network in the following literature for short), which is shown in Fig.2.

In Fig.2, n represents the number of characters of the input text segment. c_k represents input character at time k encoded using embeddings of characters through table lookup. The hidden layer s_k maintains the past and current information. The hidden activations of the last step s_n could be considered as the representation of the whole text segment. s_n is used as the input to the classification layer. y produces a probability distribution over the binary labels. Each layer is also represents a set of neurons. Layers are also connected with weights denoted by the matrices $\mathbf{U}$, $\mathbf{W}$, and $\mathbf{V}$. The values in the hidden and output layers are calculated as follows:

$$\begin{aligned}\mathbf{s}_k &= f(\mathbf{U}c_k + \mathbf{W}s_{k-1}) \\ \mathbf{y} &= f(\mathbf{V}s_n)\end{aligned} \tag{1}$$

where $f(\cdot)$ is sigmoid activation function $f(z) = \frac{1}{1+exp^{-z}}$. The architecture can be unfolded as a deep feedforward network.

We define all the parameters for the stage of modelling dynamic dictionary to be trained as $\theta = (W, U, V)$. Given an input text segment, the network with parameter θ outputs the probability, $p(1|x, \theta)$, of the given text segment can be a word or not. Cross entropy criterion is used as the loss function O of the binary classification problem. The network is trained by stochastic gradient descent using *backpropagation through time* (BPTT) (Werbos, 1990). The hidden layer activation of position i at time t, $\mathbf{s}_t^i$, is:

$$\mathbf{s}_t^i = f(a_t^i), \tag{2}$$

$$a_t^i = \sum_j u_{ij} c_t^j + \sum_l w_{il} s_{t-1}^l. \tag{3}$$

The error firstly propagates from output layer to hidden layer of last time step N. The derivatives with respect to the hidden active of position i at the last time step N can be calculated as follows:

$$\delta_N^i = f'(a_N^i) \frac{\partial O}{y} v^i, \tag{4}$$

where v^i represents the weight of hidden-output connection and the activation of the output layer y. The gradients of hidden layer of previous time steps can be recursively computed as:

$$\delta_t^i = f'(a_t^i) \sum_j \delta_{t+1}^j w_{ij}. \tag{5}$$

Given all (suppose the number is T) the training examples (x_i, y_i), we can then define the objective function as follows:

$$J(\theta) = \sum_{i=1}^{T} logp(y^{(i)} | x^{(i)}, \theta). \tag{6}$$

To compute the network parameter θ, we maximize the log likelihood $J(\theta)$ through stochastic gradient decent over shuffled mini-batches with the Adadelta(Zeiler, 2012) update rule.

2.2 RNN-RADD

As mentioned in the previous section, features extracted from external dictionary have been empirically proved to be useful for Chinese NLP various tasks. However, since these external dictionaries are usually pre-constructed, the out-of-vocabulary problem may impact the performance. Hence, we in this work propose to use RNN to determine whether a given sequence of characters is a word or not. Then the proposed RADD is incorporated into RNN (RNN-RADD is used as the abbreviation of the combined model).

2-gram	$c_{t-2}c_{t-1}$，$c_{t-2}c_{t-1}$，$c_{t-1}c_t$ c_tc_{t+i}，$c_{t+1}c_{t+2}$
3-gram	$c_{t-2}c_{t-1}c_t$，$c_{t-1}c_tc_{t+1}$，$c_tc_{t+1}c_{t+2}$
4-gram	$c_{t-2}c_{t-1}c_tc_{t+1}$，$c_{t-1}c_tc_{t+1}c_{t+2}$

Table 1: Illustration of the templates used to generate m_t.

RNN-RADD also follows the character based methodology. Hence, the basic units of RNN-RADD are Chinese characters. The architecture is illustrated in Fig. 1, where c_t denotes the input character at time t encoded using dense distributed representation. The hidden layer s_t also maintains the history of the character sequence. y_t denotes the probability distribution over labels. m_t represents the features generated through RADD. Following previous works, we construct a number of text segments from the contexts of the character based on pre-defined templates. The templates used in this work is shown in the Table. For an input text segment, RADD generates a binary value to indicate whether or not the text segment is a word a not. m_{tj} represents the value of the output corresponding to the jth template for the tth character. Each layer represents a set of neurons. Layers are connected with weights denoted by the matrices $\mathbf{U}$, $\mathbf{W}$, $\mathbf{V}$, $\mathbf{F}_s$, and $\mathbf{F}_y$.

The values in the hidden and output layers in the RNN-RADD can be expressed as follows:

$$\begin{aligned} \mathbf{s}_t &= f(\mathbf{U}c_t + \mathbf{F_s}m_t + \mathbf{W}s_{t-1}), \qquad (7) \\ \mathbf{y}_t &= g(\mathbf{V}s_t + \mathbf{F_y}m_t). \end{aligned}$$

Since RAD is trained separately with large scale domain independent dictionaries. In this work, the weight matrices of the RNN-RADD are updated with the similar way as RNN. The error loss function is computed via cross entropy criterion. The parameters are trained by stochastic gradient descent using BPTT. In order to speed up training process, the m_t and character embeddings are keep statistic, during the training procedure.

2.3 Learning Method

Based on the Eq.(3) and Eq.(4), Log-scale objective functions $Q(\theta)$ of RNN-RADD can be calculated as:

$$Q(\theta) = \sum_{t=1}^{T}(\eta a_{y^*_{t-1}y^*_t} + z_t^{y^*_t} - logZ_{R-CRF}).$$

To update the label transition weights, we compute gradients as follows:

$$\begin{aligned} \frac{\partial Q(\theta)}{\partial a_{ji}} &= \eta \sum_t \delta(y_{t-1} = j, y_t = i) \\ &- \eta \sum_t (\frac{\alpha^j_{t-1}\beta^i_t exp(\eta a_{ji} + z^t_i)}{\sum_j \alpha^j_t \beta^j_t}), \end{aligned}$$

where α^i_{t-1} is the sum of partial path scores ending at position $t-1$, with label i, which can be computed as follows:

$$\alpha^i_{t-1} = exp(z^i_{t-1}) \sum_j \alpha^j_{t-2} exp(\eta a_{ji}).$$

β^j_t is the sum of partial path scores starting at position t, with label j and exclusive of observation t, which can be computed as follows:

$$\beta^j_t = \sum_q \beta^q_{t+1} exp(\eta a_{jq} + z^j_{t+1}).$$

The model parameters θ are updated using stochastic gradient ascent (SGA) over the training data multiple passes.

3 Experiments

To demonstrate the effectiveness of the proposed method, we first compared the proposed RNN-based dynamic dictionary construction method against several baseline methods on the task. Then, we evaluated the performance of the proposed method on two Chinese natural language processing tasks: Chinese word segmentation, and opinion target extraction.

3.1 Experimental Settings

To generate the distributed representations for Chinese characters, we use the method similar to Skip-ngram (Mikolov et al., 2013), which has been successfully employed in comparable tasks. However, in this work, characters were considered the basic units of data, and the toolkit was provided by the authors[1]. We used Sogou news corpus (SogouCA[2]), which consists of news articles belonging to 18 different domains published from June 2012 to July 2012, as the training data to optimize the distributed representations of Chinese characters. After several experiments on development, we decided to set the dimension of the character embedding to 200. Through several evaluations on the validation set, in both RNN-RAD and RAD, the hidden layer size is set to 50.

3.2 Learning Chinese Dynamic Dictionary

For training and testing the proposed dynamic dictionary method, we constructed a dataset by collecting words and names from publicly available resources belonging to different domains, including a Chinese dictionary[3], an English-Chinese bilingual wordlist[4], Baidu Baike[5], the Chinese Domain Dictionary[6], and the Chinese person names list[7]. After removing duplicates, the dataset contains 11,406,995 words in total. Based on the statics of the dictionary we used, about 80.6% of Chinese person names are three characters, and words with two characters comprise the majority of the normal Chinese dictionary. Since some sources contain corporation and organization names, there are also a number of words whose lengths are longer than ten characters. However, in all sources, most of the words are less than five characters.

We randomly selected 50,000 items for use as test data and an additional 50,000 items for use as development data for tuning parameters. In addition to these positive examples, for training and testing, we also needed negative examples, so we extracted bigrams, trigrams, 4-grams, and 5-grams from the SogouCA Then, we randomly extracted a number of n-grams which were not included in the collected word lists described above as negative training data. We treat these n-grams as negative results. For training, testing, and development, we randomly selected 20 million, 50,000, and 50,000 n-grams respectively.

Besides the proposed RADD method, we also evaluated some state-of-the-art supervised methods, including:

Support Vector Machine (SVM) is one of the most common supervised methods and has been successfully used for various tasks (Hearst et al., 1998). Hence, in this work, we also evaluated its performance on the same task. We used the characters as features to construct the vector representation. Since the number of Chinese characters is limited, we used all of the characters existing in the training data. We used LIBSVM to implement (Chang and Lin, 2011).

Conditional Random Fields (CRFs) were proposed by Lafferty et al. (2001) to model sequence labeling tasks. According to the description given in §2.2, an NLP task can be converted into a sequence labeling problem. Hence, we used CRF to model characters as basic features and several combination templates of them. Compared to SVM, CRF takes both richer features and the labeling sequence into consideration. CRF++ 0.58[8] was used to do the experiments.

Dynamic Convolutional Neural Network (DCNN), defined by Kalchbrenner et al. (2014), is used to model sentence semantics. The proposed method can handle input sequences of varying length, so we adopted their method by using the embeddings of characters as input. The toolkit we used in this work is provided by the authors[9].

Recursive Autoencoder (RAE) (Socher et al., 2011), is a machine learning framework for representing variable sized words with a fixed length vector. In this work, we used greedy unsupervised RAE for modeling sequences of Chinese characters. The toolkit was provided by the authors [10]. Then, SVM was used to do the binary classification based on the generated vectors.

Table 3.2 illustrates the results of the different methods on this task. From the results, we see that the proposed method obtains the best perfor-

[1] https://code.google.com/p/word2vec/

[2] http://www.sogou.com/labs/dl/ca.html

[3] http://download.csdn.net/detail/logken/3575376

[4] https://catalog.ldc.upenn.edu/LDC2002L27

[5] http://baike.baidu.com

[6] http://www.datatang.com/data/44250/

[7] http://www.datatang.com/data/13482

[8] http://crfpp.googlecode.com/svn/trunk/doc/index.html

[9] http://nal.co/DCNN

[10] http://www.socher.org/

Methods	P	R	F1
SVM	82.27%	84.74%	83.49%
CRF	80.81%	86.82%	83.71%
DCNN	86.86%	86.55%	86.71%
RAE	84.77%	85.45%	85.11%
RADD	**89.74%**	**91.00%**	**90.39%**

Table 2: Comparison of different methods on the dynamic dictionary construction task.

mance among all of the approaches. DCNN, RAE, and RADD outperform SVM and CRF, which use characters as features. One possible reason is that the character representations are more powerful in capturing morphology than characters only. Another advantage of the deep learning framework is that it can be easily trained and makes feature engineering efforts unnecessary.

We also note that although DCNN can capture word relations of varying size in modelling sentences, RADD achieves better performance on the task of learning the morphology of Chinese. One possible interpretation is that although the relations between words in a given sentence can be well captured by DCNN, relations usually exist between nearby characters hence the recurrent network is more appropriate for the task. Moreover, RADD is much easier to implement and is more efficient than DCNN.

Fig. 3 shows the performance of RADD with different character embedding dimensions and hidden layer sizes. From the figure, we see that RADD achieves the best result when the hidden layer size is larger than 200. We also observe that RNN can achieve the highest performance with many different parameters. This means that we can easily find optimal hyper parameters.

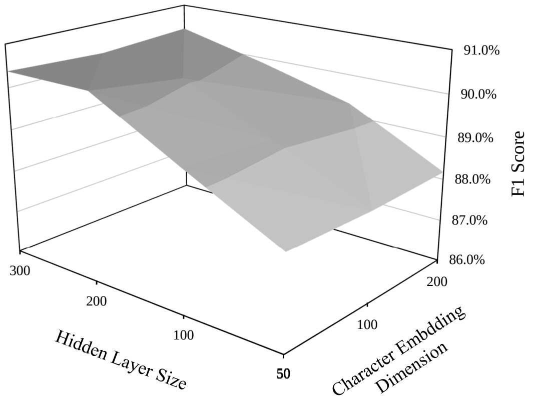

Figure 3: The results of RAD with different character embedding dimension and hidden layer size.

3.3 Experimental Results

3.3.1 Abbreviation Generation

The task of generating entity abbreviations involves producing abbreviated equivalents of the original entities. For example, 北大 is an abbreviation of 北京大学 (Peking University). Previous methods usually formulate the task as a sequence labeling problem and model it using character features (Yang et al., 2009; Xie et al., 2011). Although Chen et al. (2013) proposed to use Markov logic networks (MLN) (Richardson and Domingos, 2006) to combine local and global constraints, the morphology of Chinese was rarely considered.

In this work, we report the performance of "RNN-RADD", which takes the dynamic Chinese dictionary into consideration, on the dataset constructed by Chen et al. (2013). The dataset contains 50,232 entity abbreviation pairs. They also reported the performance achieved by their method on the dataset. We follow the strategy used by Chen et al. (2013) to generate training and test data. 75% of randomly selected pairs are used for training data, 5% for development, and the other 20% are used for testing purposes.

For comparison, we also report results achieved by the state-of-the-art methods. Yang et al. (2009) transferred the abbreviation generation method into a sequence labeling problem and proposed to use CRF to model it with several linguistic features. Chen et al. (2013) introduced local and position features and proposed to use MLN to achieve the task. We directly reference and report the results achieved by these methods on the dataset.

Table 3.3.1 shows the relative performances of the different methods. "SVM" and "RNN" denote the results of SVM and RNN on the sequence labeling problem, respectively. From the results, we see that RNN-RADD achieves the best result among all the methods. The relative improvement

Methods	Accuracy
CRFs-Yang (Yang et al., 2009)	39.70%
CRFs-LF+DPF (Chen et al., 2013)	40.60%
MLN (Chen et al., 2013)	56.80%
SVM	40.00%
RNN	60.65%
RNN-RADD	**65.98%**

Table 3: Performance of different methods on abbreviation generation task. CRFs-Yang represents the method and feature sets proposed by Yang et al. (2009). CRF-LF+DPF denotes the local and position features introduced by Chen et al. (2013). MLN represents the method incorporating local and global constraints with MLN.

of it over the previous best result achieved by MLN is about 16.2%. Comparing the performance of RNN-RADD with RNN, we also observe that the dynamic dictionary of Chinese can benefit the abbreviation generation task. The relative improvement is approximately 7.3%.

Fig. 4 shows the values of log-scale objective function of RNN and RNN-RADD during training on the data set. From this figure, we can conclude that the RNN based dynamic dictionary can benefit the task. Although additional feature vector m_i is included, the absolutely value of objective function is lower than its of RNN. It can in some degree demonstrate the effectiveness of the proposed method.

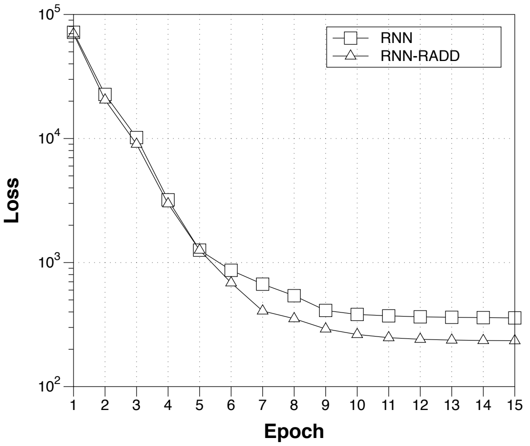

Figure 4: Comparison of RNN and RNN-RADD during training on the abbreviation data set. The vertical axis is the value of log-scale objective functions. Horizontal axis is the number of epochs during training.

	Sentences	Targets
Training	59,786	40,227
Test	11,829	8,034
Development	4,061	2,673

Table 4: Statistics of the dataset used for the opinion target extraction task.

3.3.2 Opinion Target Extraction

Opinion target extraction is a key subtask in the fine-grained opinion mining problem. The goal of it is to identify the items which opinions are expressed on from the given sentences. For example:

The *image quality* of the camera is amazing.

The "*image quality*" is the opinion target of the sentence. Previous methods studied the problem from different perspectives using supervised and unsupervised methods. Syntactic structure constituent is one of the most common assumptions used by previous works (Popescu and Etzioni, 2007; Qiu et al., 2009; Wu et al., 2009; Xu et al., 2014). Since these works usually use character level features, meaningless text segments are one of the major error types. Therefore, we integrate the dynamic Chinese dictionary into this method to detect and discard meaningless text segments.

To evaluate the proposed method, we used a dataset containing more than 6,000 reviews, which contains 75,676 sentences, about vehicles. The opinion target and opinion words were manually labeled. About 80% of the whole dataset is randomly selected for training. 15% and 5% reviews are selected as the test and development datasets respectively. Details of the data are listed in Table 3.3.2.

The task can also be modelled by sequence labelling problem. Hence, besides the proposed RNN-RADD method, we also evaluated some state-of-the-art supervised methods, including: CRF, SVM, and RNN. We used SVM and CRF under the character-based methodology for comparison. RNN is based on the character level embeddings. Table 3.3.2 shows the results of the different methods on the opinion target extraction task. From the results, we can see that, the proposed method RNN-RADD achieve the best performance in F1 score. Comparing the results of RNN with RNN-RADD, we see that the proposed dynamic dictionary

Methods	P	R	F1
CRF	71.1%	77.5%	74.2%
CRF+D	72.5%	74.3%	73.4%
SVM	77.2%	74.9%	76.0%
SVM+D	78.1%	74.3%	76.2%
RNN	79.5%	**81.7%**	80.6%
RNN-RADD	**85.5%**	81.5%	**83.4%**

Table 5: Results of different methods on the opinion target extraction task.

method can benefit the RNN based method. The error reduction achieved by its incorporation is about 11.4%. From the results of CRF and CRF+D, we can observe that dictionary is not always usefulness. We think that the main reason that the dictionary may bring too much conflict. From the results of CRF and RNN, we can see that similar to the Chinese word segmentation task, methods using character dense representations can usually achieve better performance than character based methods.

4 Related Work

Although dictionary can be manually constructed, it is a time-consuming work. Moreover, these manually constructed dictionaries are usually updated only occasionally. It would take months before it could be updated. Hence, automatic dictionary construction methods have also been investigated in recent years. Chang and Su (1997) proposed an unsupervised iterative approach for extracting out-of-vocabulary words from Chinese text corpora. Khoo (Khoo et al., 2002) introduced a method based on stepwise logistic regression to identify two-and three-character words in Chinese text. Jin and Wong (2002) incorporated local statistical information, global statistical information and contextual constraints to identify Chinese words. For collecting Thai unknown words, Haruechaiyasak et al. (2006) proposes a collaborative framework for achieving the task based on Web pages over the Internet.

Except these unsupervised methods, there have been other approaches requiring additional information or selective input. Yarowsky and Wicentowski (2000) proposed to use labeled corpus to train a supervised method for transforming past-tense in English. Rogati et al. (2003) introduced a stemming model based on statistical machine translation for Arabic. They used a parallel corpus to train the model. Luong et al. (2013) studied the problem of word representations for rare and complex words. They proposed to combine recursive neural networks and neural language models to build representations for morphologically complex words from their morphemes. Since English is usually considered limited in terms of morphology, their method can handle unseen words, whose representations could be constructed from vectors of known morphemes.

However, most of the existing Chinese dictionary construction methods focused on find out-of-vocabulary words from corpus. In this paper, we propose to transfer the dictionary construction problem to classification task and use a modified recurrent neutral network to directly model whether a given sequences of characters is a word or not.

5 Conclusion

In this work, we studied the problem of dynamic dictionary which tries to determine whether a sequence of Chinese characters is a word or not. We proposed a deep feed forward network architecture (RADD) to model the problem and integrated it into RNN method. To train the model and evaluate the effectiveness of the proposed method, we constructed a dataset containing more than 11 million words. By applying the proposed combined method to two different Chinese NLP tasks, we can see that it can achieve better performance than state-of-the-art methods. Comparing to the previous methods, the number of hyper parameters of the proposed method RNN-RADD is small and less feature engineering works are needed. In the future, we plan to integrate the dynamic dictionary into the term construction model in information retrieval.

6 Acknowledgement

The authors wish to thank the anonymous reviewers for their helpful comments. This work was partially funded by National Natural Science Foundation of China (No. 61532011, 61473092, and 61472088), the National High Technology Research and Development Program of China (No. 2015AA015408).

References

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

Jing-Shin Chang and Keh-Yih Su. 1997. An unsupervised iterative method for chinese new lexicon extraction. *Computational Linguistics and Chinese Language Processing*, 2(2):97–148.

Huan Chen, Qi Zhang, Jin Qian, and Xuanjing Huang. 2013. Chinese named entity abbreviation generation using first-order logic. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*.

Choochart Haruechaiyasak, Chatchawal Sangkeettrakarn, Pornpimon Palingoon, Sarawoot Kongyoung, and Chaianun Damrongrat. 2006. A collaborative framework for collecting thai unknown words from the web. In *Proceedings of the COLING/ACL*.

Marti A. Hearst, ST Dumais, E Osman, John Platt, and Bernhard Scholkopf. 1998. Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28.

Honglan Jin and Kam-Fai Wong. 2002. A chinese dictionary construction algorithm for information retrieval. *ACM Transactions on Asian Language Information Processing (TALIP)*.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proceedings of ACL*.

Christopher SG Khoo, Yubin Dai, and Teck Ee Loh. 2002. Using statistical and contextual information to identify two-and three-character words in chinese text. *Journal of the American Society for Information Science and Technology*, 53(5):365–377.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*.

Xiaohua Liu, Ming Zhou, Furu Wei, Zhongyang Fu, and Xiangyang Zhou. 2012. Joint inference of named entity recognition and normalization for tweets. In *Proceedings of ACL*.

Minh-Thang Luong, Richard Socher, and Christopher D. Manning. 2013. Better word representations with recursive neural networks for morphology. In *CoNLL*, Sofia, Bulgaria.

Catherine McBride-Chang, Hua Shu, Aibao Zhou, Chun Pong Wat, and Richard K Wagner. 2003. Morphological awareness uniquely predicts young children's chinese character recognition. *Journal of Educational Psychology*, 95(4).

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

Fuchun Peng, Fangfang Feng, and Andrew McCallum. 2004. Chinese segmentation and new word detection using conditional random fields. In *Proceedings of the 20th international conference on Computational Linguistics*.

Ana-Maria Popescu and Orena Etzioni. 2007. Extracting product features and opinions from reviews. In *Natural language processing and text mining*, pages 9–28. Springer.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2009. Expanding domain sentiment lexicon through double propagation. In *IJCAI*, volume 9, pages 1199–1204.

Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Machine Learning*, 62(1-2):107–136.

Monica Rogati, Scott McCarley, and Yiming Yang. 2003. Unsupervised learning of arabic stemming using a parallel corpus. In *Proceedings of ACL 2003*.

Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the EMNLP '11*.

Paul J Werbos. 1990. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*.

Youzheng Wu, Jun Zhao, Bo Xu, and Hao Yu. 2005. Chinese named entity recognition based on multiple features. In *Proceedings of HLT/EMNLP*.

Yuanbin Wu, Qi Zhang, Xuangjing Huang, and Lide Wu. 2009. Phrase dependency parsing for opinion mining. In *Proceedings of EMNLP*.

Li-Xing Xie, Ya-Bin Zheng, Zhi-Yuan Liu, Mao-Song Sun, and Can-Hui Wang. 2011. Extracting chinese abbreviation-definition pairs from anchor texts. In *Machine Learning and Cybernetics (ICMLC)*.

Liheng Xu, Kang Liu, Siwei Lai, and Jun Zhao. 2014. Product feature mining: Semantic clues versus syntactic constituents. In *Proceedings of the 52nd ACL*.

Dong Yang, Yi-cheng Pan, and Sadaoki Furui. 2009. Automatic chinese abbreviation generation using conditional random field. In *Proceedings of NAACL 2009*.

David Yarowsky and Richard Wicentowski. 2000. Minimally supervised morphological analysis by multimodal alignment. In *Proceedings of ACL*.

Matthew D Zeiler. 2012. Adadelta: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Lei Zhang, Bing Liu, Suk Hwan Lim, and Eamonn O'Brien-Strain. 2010. Extracting and ranking product features in opinion documents. In *Proceedings of the 23rd international conference on computational linguistics: Posters*.