2017 28th International Workshop on Database and Expert Systems Applications

Opinion Expression Detection via Deep Bidirectional C-GRUs

Xiaoxia Xie
School of Electrical Science and Engineering
National University of Defense Technology
Changsha, Hunan,China (0731)410073
Email: xiexiaoxia@nudt.edu.cn

***Abstract*—The ability to accurately detect opinion expression in a document is an essential and fundamental task in opinion mining. In this work, we consider opinion expression detection as a sequence labeling task. We describe deep neural network frameworks that consist of convolutional neural networks (CNNs) and bidirectional gated units (Bi-GRUs). CNNs are capable of capturing local features in a sequence, while Bi-GRUs, a type of recurrent neural network (RNN) variant, are able to extract features from sequence data. The properties of these two networks provide the framework to effectively detect opinion expression. Experimental results show that our methods significantly outperform traditional methods like conditional random field (CRF) and previous state-of-the-art deep RNN methods.**

Keywords: opinion mining, opinion expression detection, sequence labeling, recurrent neural network, convolution neural network.

I. Introduction

Opinion mining, also called sentiment analysis, aims to identify the sentiment expressed in a document and is currently becoming an increasingly popular area of investigation. Opinion expressions refer to the way people express emotions, feelings, attitudes, and interests using natural language. Opinion expression detection is a fundamental and essential task in opinion mining, which involves figuring out opinion holders that express opinion and emotion.

Wiebe et al. [1] proposed two types of opinion expressions: direct subjective expressions (DSEs) and expressive subjective expressions (ESEs). DSEs represent both subjective speech events and explicitly mentioned private states, while ESEs consist of tokens that express emotion or sentiment in an indirect or implicit way.

Therefore, opinion expression detection can be considered a linguistic sequence labeling problem [2]. As shown in Table I, each token in the sentence is tagged with “BIO” letters which indicate the beginning, inside, or outside of DSE or ESE, respectively. For example, the DSEs “are prepared to initiate a dialogue” and “want” express a positive attitude, while the ESE “our friends and allies” implicitly conveys feeling and attitude.

Traditional methods such as conditional random field (CRF) or HMM [1] have been successfully applied in opinion expression detection. Recently, given the advances in neural networks and deep learning, neural networks and their variants have received increasing attention, and have been applied in numerous sequence labeling problems. It appears that these types of methods surpass many traditional methods [2]. Among these neural networks, convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are two frequently used models.

CNNs perform convolutional operations on adjacent features, which enables an efficient and effective extraction of local features. Staking CNNs can mine latent information of input data. However, CNNs are not suitable for handling sequence labeling directly. In many studies [3], [4], CNNs are used to extract features at the token or phrase level, which are then sent on for further processing.

RNNs have shown superiority in a variety of sequence labeling tasks, and are even better than some traditional methods such as CRF and HMM [2] due to their ability to capture local features in a sequence. Moreover, bidirectional RNNs (Bi-RNNs) combine forward and backward sequence information, which provides a better understanding of the sequence. Gated recurrent units (GRUs) is a variant of RNN that overcome the shortcomings of RNNs. In this paper, we select Bi-GRUs as our sequence model.

Motivated by the advantages of CNNs and Bi-GRUs, we propose deep neural network frameworks that combine several layers of CNNs and Bi-GRUs which benefit from their abilities to capture local features and long term memory.

The contributions of our work are: (1) developing a new deep neural network framework to sequence the labeling of text and apply it to opinion expression extraction. (2) exploring bidirectional GRUs and deep models to achieving state-of-the-art performance.

II. Related Work

This section presents a brief survey about opinion expression detection and introduces deep learning applied in natural language processing (NLP).

A. Opinion expression detection

Opinion expression detection is a crucial task in fine-grained opinion mining. Wiebe et al. [1] defined opinion expression as direct subject expression and expressive subject expression, and constructed a fine-grained opinion mining corpus named MPQA. Breck et al. [5] formulated opinion expression

2378-3915/17 $31.00 © 2017 IEEE
DOI 10.1109/DEXA.2017.40

TABLE I

AN EXAMPLE OF LABELING A SENTENCE

We	**are**	**prepared**	**to**	**initiate**	**a**	**dialogue**	**and**	**we**
O	B_DSE	I_DSE	I_DSE	I_DSE	I_DSE	I_DSE	O	O
O	O	O	O	O	O	O	O	O
want	**to**	**work**	**with**	**our**	**friends**	**and**	**allies**	**.**
B_DSE	O	O	O	O	O	O	O	O
O	O	O	O	B_ESE	I_ESE	I_ESE	I_ESE	O

detection as a sequence labeling method and achieved good performance via CRF. Choi et al. [6] applied a CRF-based approach to extract opinion expressions and to jointly assign polarities. Irsoy et al. [2] applied deep RNNs to the opinion expression detection task and achieved the state-of-the-art result.

B. Deep learning in NLP

Recently, deep learning is becoming increasingly important. Deep learning has been successfully applied in numerous natural language processing (NLP) tasks. Among the neural network models used in NLP tasks, convolutional neural network (CNN) and recurrent neural network (RNN) are the most popular [7].

a) CNNs: Due to their ability to capture local correlations of spatial or temporal structures, convolutional neural networks (CNNs) are currently the cutting edge in computer vision [8] and audio processing [9]. CNNs are prevalent neural network architectures that are used in many NLP tasks. A variety of studies use CNNs as a word-level processing tool for further modeling at the sentence or document level, such as machine translation [10], text classification [11], sentiment analysis [12] and Information Retrieval [13].However, only a few studies that use CNNs for sequence labeling tasks in NLP. Xu et al. [14] use one CNN layer to learn word features within a window context and then add a TriCRF layer for slot filling and intent detection. In this paper, CNNs are selected to extract local features of words as a means of preprocessing for other models.

b) RNN: Recurrent neural networks (RNNs), a type of neural network where connections between adjacent nodes are in one direction, are developed for processing sequence or series tasks. When viewing a sentence as a sequence of tokens, RNNs are able to achieve excellent performance in several NLP tasks. Mikolov [15] proposed an RNN-based language modeling approach, which achieved better performance than a feed-forward neural network. Li et al. [16] applied RNNs to biomedical named entity recognition and achieved better results than CRF method. Irsoy et al. [2] discussed deep bidirectional RNN models in opinion expression detection and achieved greater results than CRF and semiCRF methods. Given the excellent performance of RNNs, we tend to use RNNs and its variants as the sequence labeling model for sentiment parsing. As long as CNNs do well in extracting local features and RNNs do well in sequence processing tasks, then we attempt to combine them to produce better results.

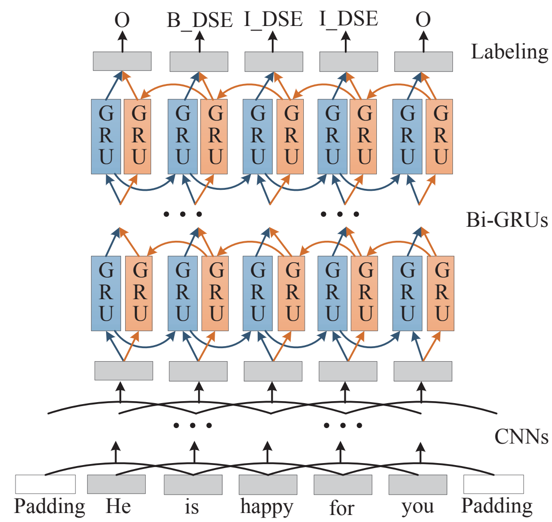

Fig. 1. The architecture of deep bidirectional C-GRUs for sequence labeling.

III. NEURAL NETWORK FRAMEWORKS

This section presents the components of our neural network frameworks. As shown in Fig. 1, a sentence is viewed as a sequence of tokens, and each token is at first embedded into a fixed-length word vector. Convolutional neural networks then extract local contextual features of each token and, in some cases, they are stacked in several layers in order to extract more latent features. The features are then fed into bidirectional GRUs that mine semantic information and extract global contextual features for each token. Finally, a labeling layer is attached to the Bi-GRUs output in order to calculate the labels for each token.

A. Convolution

It is important to consider the contextual information of a token when analyzing the meaning and corresponding label because a token may have different meanings in different contexts. Previous studies have shown that CNNs are effective at extracting local features [8], [9], so we apply CNN layers to capture contextual features of tokens after word embedding.

In general, convolution and pooling are two significant operations in CNNs. Convolutional operations extract each local feature with a number of filters that have a fixed length and slide over the inputs. Pooling operations aggregate invariance features in a region by calculating the mean (max or sum) value of features over the region. Multi-layers of alternating convolution and pooling can be used to represent features in different scales. For sequence labeling tasks, we need to keep the length of the output layer the same as the length of the input sentence. Therefore, we only use convolution step to extract local features for each word.

Specifically, in terms of sentence $S = \{t_1, t_2, ..., t_s\}$, each token t_j is represented as a d-dimensional vector $e_{t_j} \in R^d$. In convolution operations, let m be the filter length, n be the number of filters, $W_{ci} \in R^{m\times d}$ be the filter weight of the ith filter, and $c_j \in R^n$ be the feature vector of t_j after a convolutional operation. In order to avoid bias in the convolutional operation, m is an odd number in this paper. The convolution operation is computed as:

$$c_{ji} = g(W_{ci} \circ [e_{t_{j-m/2+1/2}}, ..., e_{t_j}, ..., e_{t_{j+m/2-1/2}}] + b_i), \quad (1)$$

where $\circ$ denotes element-wise multiplication, $b_i \in R$ is a bias term and g is a nonlinear active function (sigmoid function, relu or $tanh$). When $(j - m/2 + 1/2) < 1$ or $(j + m/2 - 1/2) > s$, the embedding vectors of padding are used to compute.

B. Bidirectional GRUs

Gated recurrent units (GRUs) is a famous RNN variants that overcome the gradient vanishing problem and achieve excellent performance in a variety of language processing tasks [17], [18]. Therefore, we select GRU as the recurrent unit of RNN to capture sequence information after the convolution operations. However, the output of the GRU in each step is determined by the input of the current step and previous steps, while ignoring the inputs after it. For a sentence, the label of a token often depends on the following tokens as well as the previous tokens. Therefore, we apply bidirectional GRU (Bi-GRU) to solve this problem.

Bi-GRU has two hidden RNN layers, which work independently and the output of each step is computed by the combination of the two hidden units:

$$\overrightarrow{r_j} = f(\overrightarrow{W_r} x_j + \overrightarrow{U_r}\overrightarrow{h}_{j-1} + \overrightarrow{b_r}), \quad (2)$$
$$\overrightarrow{z_j} = f(\overrightarrow{W_z} x_j + \overrightarrow{U_z}\overrightarrow{h}_{j-1} + \overrightarrow{b_z}), \quad (3)$$
$$\overrightarrow{h}'_j = g(\overrightarrow{W_h} x_j + \overrightarrow{U_h}(\overrightarrow{r_j} \odot \overrightarrow{h}_{j-1}) + \overrightarrow{b_h}), \quad (4)$$
$$\overrightarrow{h}_j = \overrightarrow{z_j}\overrightarrow{h}_{j-1} + (1 - \overrightarrow{z_j})\overrightarrow{h}'_j, \quad (5)$$
$$\overleftarrow{r_j} = f(\overleftarrow{W_r} x_j + \overleftarrow{U_r}\overleftarrow{h}_{j+1} + \overleftarrow{b_r}), \quad (6)$$
$$\overleftarrow{z_j} = f(\overleftarrow{W_z} x_j + \overleftarrow{U_z}\overleftarrow{h}_{j+1} + \overleftarrow{b_z}), \quad (7)$$
$$\overleftarrow{h}'_j = g(\overleftarrow{W_h} x_j + \overleftarrow{U_h}(\overleftarrow{r_j} \odot \overleftarrow{h}_{j+1}) + \overleftarrow{b_h}), \quad (8)$$
$$\overleftarrow{h}_j = \overleftarrow{z_j}\overleftarrow{h}_{j+1} + (1 - \overleftarrow{z_j})\overleftarrow{h}'_j. \quad (9)$$
$$h_j = \overrightarrow{V}\overrightarrow{h}_j + \overleftarrow{V}\overleftarrow{h}_j. \quad (10)$$

where $\overrightarrow{W_*}$ and $\overrightarrow{U_*}$ are forward weight matrices, $\overrightarrow{b_*}$ is forward bias vector. $\overleftarrow{W_*}$ and $\overleftarrow{U_*}$ are the backward weight matrices, $\overleftarrow{b_*}$ is backward bias vector. The $\odot$ operation denotes the element-wise vector product, and f and g are nonlinear functions. $\overrightarrow{V}$ and $\overleftarrow{V}$ are the weight matrices for combining two hidden units.

C. Labeling

The labeling layer determines the label of each token based on the output of the last Bi-GRU layer.

Specifically, we utilize a three-dimensional vector to indicate the label of each token. In terms of DSE tasks, the label l_j of token t_j can be written as:

$$y_j^l = \begin{cases} (0,0,1), where \quad l_j = B\_DSE, \\ (0,1,0), where \quad l_j = I\_DSE, \\ (1,0,0), where \quad\quad l_j = O. \end{cases} \quad (11)$$

In the labeling layer, the output Bi-GRUs at each step y_j is translated into a three-dimensional vector and normalized by a softmax function to obtain the label vector $\hat{y}_j^l$:

$$\hat{y}_j^l = softmax(W_l y_j + b_l). \quad (12)$$

where W_l is the weight matrix and b_l is the bias vector.

Similarly, in terms of ESE tasks, the label l_j of token t_j can also be written in the form of Eq. (11), and outputs of the labeling layer can also be calculated in the form of Eq. (12).

D. Objective function

The objective function, also called the loss function, is an equation that should be optimized in neural networks. In general, the goal of training a neural network framework is to minimize its objective function.

A common objective function for multi-class classification is categorical cross-entropy, which is also called multi-class log-loss. Categorical cross-entropy is used in our work and can be written as follows:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_j \log \hat{y}_j, \quad (13)$$

TABLE II
RESULTS FOR DSE TASKS

CNN depth	GRU depth	Precision		Recall		F-score	
		Bin.	Prop.	Bin.	Prop.	Bin.	Prop.
1	1	71.99	**68.27**	71.00	64.07	71.36	65.96
2	1	72.18	67.34	71.82	65.90	71.92	66.54
3	1	71.68	67.36	72.03	66.21	71.80	66.72
1	2	72.29	67.48	70.88	64.63	71.48	65.92
2	2	**72.66**	68.06	71.75	65.96	**72.12**	**66.90**
3	2	71.71	67.12	**72.04**	**66.36**	71.82	66.67

TABLE III
RESULTS FOR ESE TASKS

CNN depth	GRU depth	Precision		Recall		F-score	
		Bin.	Prop.	Bin.	Prop.	Bin.	Prop.
1	1	63.70	**54.83**	70.84	58.23	66.95	56.24
2	1	64.51	53.21	71.46	61.32	67.61	56.62
3	1	62.98	51.80	**73.24**	**62.89**	**67.65**	**56.66**
1	2	64.85	52.63	70.59	60.33	67.50	55.94
2	2	65.35	53.77	70.11	59.83	67.47	56.37
3	2	**66.08**	54.31	69.04	59.13	67.35	56.15

TABLE IV
COMPARISON BETWEEN PREVIOUS METHODS AND OURS

Tasks	**GRU depth**	**Precision**		**Recall**		**F-score**	
		Bin.	Prop.	Bin.	Prop.	Bin.	Prop.
DSE	CRF + vec	**82.43**	**74.97**	55.67	49.47	66.44	59.59
	semiCRF + vec	71.98	66.00	68.13	60.96	69.91	63.30
	3-layers RNN	69.12	65.56	**74.69**	**66.73**	71.72	66.01
	2-layers CNN + 2-layers GRU	72.66	68.06	71.75	65.96	**72.12**	**66.90**
ESE	CRF + vec	69.84	**57.15**	54.38	44.67	61.01	50.01
	semiCRF + vec	**70.82**	53.75	61.59	52.72	65.73	53.10
	3-layers RNN	60.50	52.04	**76.02**	61.71	67.18	56.26
	3-layers CNN + 1-layers GRU	62.98	51.80	73.24	**62.89**	**67.65**	**56.66**

where N indicates the number of samples, and M is the sequence.

IV. EXPERIMENTS

A. Data set

Data set MPQA 1.2 is used in this paper, which was created by Wiebe et al. [1] and contains 535 news articles that are made up of 11,111 sentences. All sentences are tagged at the token level. In our experiments, we split the data set into 10-folds for cross validation.

B. Result and Discussion

1) Different deep neural frameworks: MPQA labels DSE and ESE in separate files, so we perform experiments on DSE and ESE sequence labeling, respectively. By changing the depth of CNNs and RNNs, we obtain different deep neural network frameworks. Table II and III show the results of DSE and ESE, respectively, in terms of different frameworks.

As shown in Table II, we vary the number of CNN layers from 1 to 3, and RNN layers from 1 to 2. The 2-layer CNN and 2-layer Bi-GRU provide the best F-score for the DSE task. With an increase in the CNN layers, the recall shows an upward trend. This may be evidence that CNN layers can capture adjacent information in a sequence.

Results shown in Table III indicate that 3-layer CNN and 1-layer Bi-GRU provide the best F-score for the ESE task. With an increase in the Bi-GRU layers, the recall and F-score show a slight decrease. Interestingly, with an increase in the CNN layers, we find that F-score increases when the layer of Bi-GRU is 1, and decreases when it is 2.

Overall, given the great learning ability of deep neural networks, these results seem to be relatively close and stable. In addition, batch normalization and dropout guarantee our frameworks from over-fitting.

2) Comparison to previous results: Irsoy et al. [2] performed the experiment on the same data set, where they stack RNNs into deep neural networks to yield state-of-the-art results. They also compare their models to traditional models such as CRF and semi-CRF, but show that deep RNN models are superior.

Table IV shows a comparison between our models and previous models. In terms of F-score, our methods yield new state-of-the-art results in both DSE and ESE detection tasks.

Conditional random field (CRF) with word vectors produces very high precision, but also very low recall, leading to a low F-score in both DSE and ESE tasks. Unlike CRF, which operates at the token level, semi-CRF operates at the phrase level and uses word vectors to represent phrases. Semi-CRF with word vectors overcomes the low recall problem of CRF and reaches a higher F-score. As mentioned above, 3-layer RNN surpasses these CRF-based methods in F-score due to great performance in recall. Compared to these methods, the methods proposed in this paper obtain superior results.

V. Conclusions and Future Work

We presented deep neural network frameworks based on CNNs and bidirectional GRUs, which can be successfully applied in opinion expression detection tasks. The methods proposed in this paper surpass traditional methods such as CRF and semi-CRF. Compared to previous state-of-the-art methods, experimental results show the superiority of our methods. In future work, we intend to apply our models to other opinion mining tasks and apply other models to sentiment analysis tasks.

References

[1] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Language resources and evaluation*, vol. 39, no. 2-3, pp. 165–210, 2005.

[2] O. Irsoy and C. Cardie, "Opinion mining with deep recurrent neural networks," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, Conference Proceedings, pp. 720–728.

[3] J. P. Chiu and E. Nichols, "Named entity recognition with bidirectional lstm-cnns," *arXiv preprint arXiv:1511.08308*, 2015.

[4] X. Ma and E. Hovy, "End-to-end sequence labeling via bi-directional lstm-cnns-crf," *arXiv preprint arXiv:1603.01354*, 2016.

[5] E. Breck, Y. Choi, and C. Cardie, "Identifying expressions of opinion in context," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, 2007, Conference Proceedings, pp. 2683–2688.

[6] Y. Choi and C. Cardie, "Hierarchical sequential learning for extracting opinions and their attributes," in *Proceedings of the ACL 2010 conference short papers*. Association for Computational Linguistics, 2010, Conference Proceedings, pp. 269–274.

[7] C. Zhou, C. Sun, Z. Liu, and F. Lau, "A c-lstm neural network for text classification," *arXiv preprint arXiv:1511.08630*, 2015.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, no. 2, p. 2012, 2012.

[9] Z. W. Huang, W. T. Xue, and Q. R. Mao, "Speech emotion recognition with unsupervised feature learning," *Frontiers of Information Technology & Electronic Engineering*, vol. 16, no. 5, pp. 358–366, 2015.

[10] F. Meng, Z. Lu, M. Wang, H. Li, W. Jiang, and Q. Liu, "Encoding source language with convolutional neural network for machine translation," *arXiv preprint arXiv:1503.01838*, 2015.

[11] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[12] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, Conference Proceedings, pp. 1422–1432.

[13] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil, "Learning semantic representations using convolutional neural networks for web search," in *Proceedings of the companion publication of the 23rd international conference on World wide web companion*. International World Wide Web Conferences Steering Committee, 2014, Conference Proceedings, pp. 373–374.

[14] P. Xu and R. Sarikaya, "Joint intent detection and slot filling using convolutional neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2014, Conference Proceedings.

[15] T. Mikolov, M. Karafit, L. Burget, J. Cernock, and S. Khudanpur, "Recurrent neural network based language model," in *INTERSPEECH*, 2010, Conference Proceedings, pp. 1045–1048.

[16] L. Li, L. Jin, Z. Jiang, D. Song, and D. Huang, "Biomedical named entity recognition based on extended recurrent neural networks," in *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, Nov 2015, pp. 649–652.

[17] K. Cho, B. Van Merrinboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[18] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.