

Sentiments Analysis of Reviews based on ARCNN model

Xiaoyu Xu¹, Ming Xu¹, Jian Xu¹, Ning Zheng¹, Tao Yang^{2*}

¹School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China

²Key Lab of the Third Research Institute of the Ministry of Public Security

yangtao@stars.org.cn

Abstract. The sentiments analysis of product reviews is designed to help customers understand the status of the product. The traditional method of sentiments analysis relies on the input of a fixed feature vector which is performance bottleneck of the basic codec architecture. In this paper, we propose an attention mechanism with BRNN-CNN model, referring to as ARCNN model. In order to have a good analysis of the semantic relations between words and solves the problem of dimension disaster, we use the GloVe algorithm to train the vector representations for words. Then, ARCNN model is proposed to deal with the problem of deep features training. Specifically, BRNN model is proposed to investigate non-fixed-length vectors and keep time series information perfectly and CNN can study more connection of deep semantic links. Moreover, the attention mechanism can automatically learn from the data and optimize the allocation of weights. Finally, a softmax classifier is designed to complete the sentiment classification of reviews. Experiments show that the proposed method can improve the accuracy of sentiment classification compared with benchmark methods.

1. Introduction

Since entering the 21st century, the Internet has gradually become an indispensable part of people's lives. With more and more people join the network, a large number of network information will be generated. This information contains both the objective facts that already exist and the subjective emotions, and the text is one of the most important ways in which the information presents to people.

Sentiment analysis is an important research in the natural language processing filed. To do this research, the first step is to convert the text and feature into a digital sequence. It is common to use bag of words or n-grams to represent text [1]. However, these methods are not only has a high-dimensional representation of word, but ignore the whole words order in the text, which limits the functionality of the learning models based on such representation. In recent years, the distributed representation methods have brought new solutions to learn better representation of text. The most commonly used methods are Word2Vec [2] and GloVe [3] that have a good analysis of the semantic relations between words and solves the problem of dimension disaster.

Based on these represent, the earliest studies of sentiment analysis focus on machine learning based methods [4, 5] and lexicon based method. The performance of machine learning methods such as Support Vector Machine [5], Naïve Bayes, Maximum Entropy model, Random Walk model strongly depends on the quality of the extracted features, and the performance of lexicon based method relied heavily on the quality of emotion lexicon [6]. These methods can learn the shallow features of a single word accurately, but cannot learn the semantic relations between words and words in the text. In

addition, many researchers also use deep learning methods to deal with the sentiments analysis, such as the convolution neural network (CNN) [7] and recurrent neural networks (RNN) [8]. CNN treats each words fairly and uses sliding windows which also called filters to do the feature mapping, then a pooling is to get the fixed length output. Unlike the CNN, RNN takes into account the order of words and could better deal with the text. Although some models based on the deep neural network can learn the deep semantic relation between the word and the word and the sentence and the sentence, it does not highlight the characteristics of the single word.

In view of the above problems, this paper mainly uses the deep learning methods to study the sentiment analysis of the reviews. We build the bidirectional recurrent neural network to handle the input with variable comment text. And inspired by recent success of attention mechanism [9], we combine the attention mechanism with BRNN to focus on the specific scope of the text. This is important for the semantic extraction of text and its visibility can directly explain the importance of each word to the sentiment analysis. Then, we add a convolution process which can better handle local information. We have evaluated our proposed method with benchmark experiments. Results reveal in Section 4 and show that it is effective to use the ARCNN to analyse the sentiment.

The rest parts of this paper are organized as follow. The following section reviews the related work. Next, the method for sentiment analysis is described in Section 3. And Section 4 reports the results of our experiments. Finally, Section 5 draws a conclusion.

2. Related work

In recent years, the neural network based sequence model has been a great success in text processing, especially in comparison with the BOW (bag of words) model, showing significant advantages. CNN has made a lot of outstanding achievements in many aspects, such a parsing, search word retrieval, sentence model and other traditional natural language work. CNN uses the local spatial correlation between layer and layer, and the neurons of each adjacent layer is only connected with the closer previous neuron node, which greatly reduces the architecture parameters size of the Neural Network [10]. In addition to CNN, RNN also been used in the NLP [11, 12]. In the field of machine translation, the introduction of RNN makes the dialogue model have made new progress. Sutskever et al. [13] proposed an end-to-end sequence learning method that minimizes the computational complexity of sequence structures. In the task of sentiment analysis, both of CNN and RNN can play a good role. Kim et al. [7] used a simple convolution neural network and word vectors as the input of model. Liang et al. [8] extended LSTM to recursive neural network to capture the deeper level of semantic syntax information. Wen et al. [14] proposed a RCNN with Highway network to learning text representation for sentiment analysis.

And since Denil et al. [15] proposed attention mechanism, the deep learning model which relies on this mechanism has been successfully applied to a variety of NLP tasks. In 2014, Bahdanau et al. [16] used the attention model to adjust the input and output sequences of machine translations and achieved good results. In 2015, Xu et al. [17] proposed an attention-based model to automatically learn to describe the contents of the images. In the same year, Sharma et al. [18] divided the attention model into two categories, soft attention and hard attention, and the method can be used to learn the weight of attention.

3. Method

The figure 1 gives architecture of our method. Next we introduce it in several major modules.

3.1. Input

For the traditional word representation, such as one-hot representation [1], the biggest problem is that it can't analyse the semantic relation between words, and its dimension is particularly large. Recent research has demonstrated distributed representation [2, 3] of word embedding is more powerful. A word can be represented by a vector as follows:

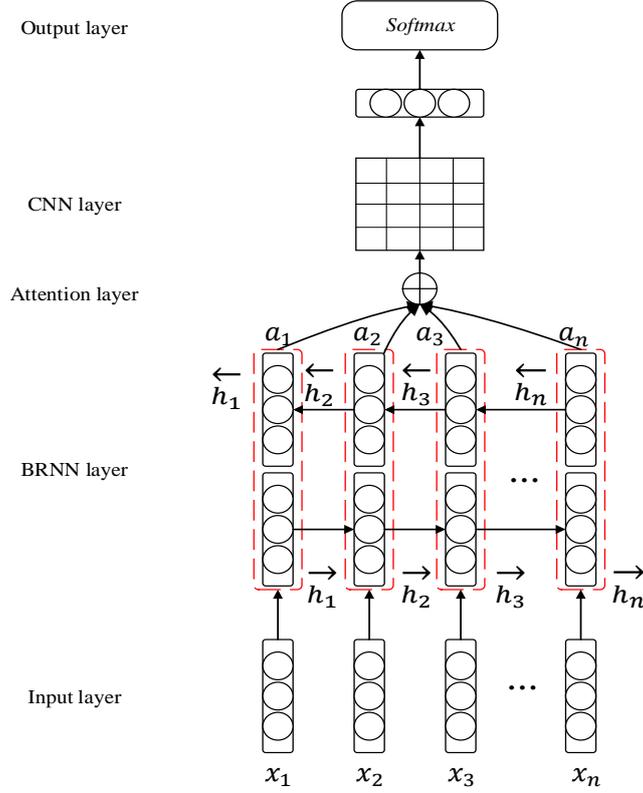


Figure 1. The architecture of the proposed model

$$W = Lx \quad (1)$$

where $x \in R^{|N|}$ is a one-hot vector and $L \in R^{d \times |N|}$ is a word-representation matrix. In fact, it is a mapping from high dimension to low dimension. In this paper, we use *Glove* embedding proposed by [3]. The model is trained by minimize the loss function:

$$J(\theta) = \frac{1}{2} \sum_{i,j=1}^W f(P_{ij})(u_i^T v_j - \log P_{ij}) \quad (2)$$

3.2. BRNN and Attention mechanism

The recurrent neural network is a special feedforward neural network, which is very effective for modelling a time-dependent sequence. The characteristics of its loop structure make the output of each moment associated with the output of the previous time, which is very suitable for dealing with variable-length text, and can well express the correlation between nodes in the sequence. Most of the existing models compile the input sequence into a fixed-length vector, which ignores the characteristics of the dimension that is important for the expression of the sentence and severely limits the resulting semantic vector. The advantage of the recurrent neural network is that its recurrent structure can handle non-fixed-length vectors and can keep the time series information.

The single RNN derives the characteristics of the current word based on the output characteristics of the previous word, which takes into account only one side of the text and does not take into account the effect of the next word on the current word. In order to make full use of all the information contained in the text sequence, this paper uses the Bidirectional Recurrent Neural Networks (BRNN) [19] to encode the input sequence. The network in parallel two time directions, and at every moment,

the past information and feature information can be used directly. The hidden layer state is calculated as follows:

$$\vec{h}_t = f(\vec{U}x_T + \vec{V}h_{t-1} + \vec{b}) \quad (3)$$

$$\overleftarrow{h}_t = f(\vec{U}x_T + \vec{V}h_{t+1} + \vec{b}) \quad (4)$$

where the f is an activation function, b is the offset and U, V are the shared weight parameters of the entire neural network. In addition, $\vec{h}_t, \overleftarrow{h}_t$ represent the forward hidden state and backward hidden state respectively. We represent the final state at position t with the concatenation of \vec{h}_t and \overleftarrow{h}_t .

$$h_t = \vec{h}_t \parallel \overleftarrow{h}_t \quad (5)$$

After the BRNN, we consider the attention mechanism. The central idea of the attention mechanism is to let the model automatically find information about the relevant parts (usually as vector sequences) from the stored memory and use the information to get the target results. The advantage of using the attention mechanism is that, firstly, a complete review usually consists of many factors, and the model can learn how to effectively integrate the various factors into a whole, rather than simply translating information from one mode to another mode.

In this part, we use a 2-layer feedforward neural network [16] to train attention scores and the number of memory cells is equal to the BRNN hidden unit. The attention mechanism gives an attention score $e_{i,t}$ to each word t in the sentence i .

$$e_{i,t} = g(Wh_t, c) \quad (6)$$

where the g is an activation function. And then, the weight probability $a_{i,t}$ of each h_t is computed by

$$a_{i,t} = \frac{\exp(e_{i,t})}{\sum_{k=1}^T \exp(e_{i,k})} \quad (7)$$

3.3. CNN and Output

Convolutional layers can encode significant information about input data with significantly fewer parameters than other deep learning. And as mentioned in the previous section, we use the attention model to connect CNN and BRNN to obtain the better feature representations.

For the CNN, given a window size w , a filter is seen as a weight matrix $f = (f_1, f_2, \dots, f_w)$ (f_i is a column vector), and the bias is seen as b . A feature map can be obtained by an activation function f . Each output map could be a value combining by multiple filters, and it can be calculated as follows:

$$z_i = f\left(\sum_{j=0}^{w-1} f_{j+1}u_{j+i} + b\right) \quad (8)$$

where z_i and u_i is an element in the output and input, respectively. And we use the max pooling to further abstract the features generated from the convolutional layer.

The output of the above layers is regarded as features of sentences for sentiment classification. At the end of the model, there is a full-connection *softmax* classifier which gets the output of each category. We use the $e p(i|x, \theta)$ to denote the probability of the review being class i , where θ is the parameters of model:

$$p(i|x, \theta) = \frac{\exp(y_i)}{\sum_{j=1}^k \exp(y_j)} \quad (9)$$

4. Experiments

4.1. Dataset

In this section, experiments are conducted to evaluate the performance of the proposed algorithm. The dataset used in the experiments comes from the Kaggle competition, and it contains 25000 movie reviews. Each review is labelled by human with 1 represents positive sentiment and 0 represents negative sentiment. Data sample showed as table 1. We randomly selected 5000 reviews as the test set, leaving the 20000 as the training set.

4.2. Results and evaluation

In this section, we designed three experiments:

- BRNN: only use the Bidirectional Recurrent Neural Network to do sentiments analysis;
- ARNN: combine the attention mechanism with BRNN to do sentiments analysis;
- ARCNN: combine the attention mechanism with BRNN-CNN to do sentiments analysis.

Table 1. Data sample

Positive	Tt is a fantastic film! I've not laughed so much when watching a film for ages!
	This is a wonderful story of a woman's bravery, courage and extreme loyalty.
Negative	This is an awful film. Yea the girls are pretty but it's not very good.
	I do not recommend this movie , because it's inaccurate and misleading

We use the GloVe algorithm to train the vector representations for words and adopt softmax output to make prediction. The word embedding is pre-trained in a corpus of about 400000 in size, and the dimension of word vector is set to 100. For our model, ARCNN, we use 'RMSprop' [20] as training method with batch size of 50, the BRNN hidden layer dimension of 100, and the CNN filters number of 64.

In order to evaluate the effectiveness of our method for sentiment analysis of reviews, we use the accuracy as evaluate metric. The accuracy reflects the ability of the classifier system to judge the whole sample and is defined as:

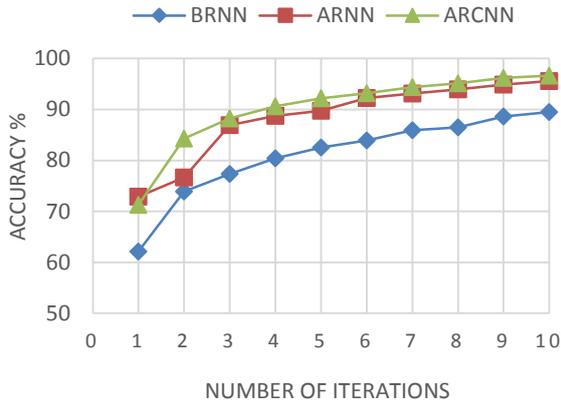
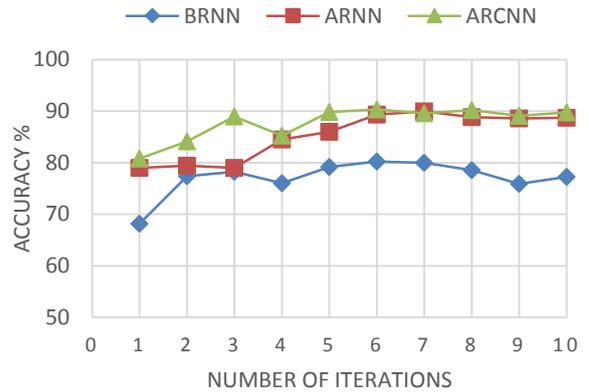
$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (10)$$

where the TP is the number of reviews that are correctly divided into positive review, i.e. the number of the reviews that are classified as positive and its real category is positive too. The TN is the number of reviews that belong to negative category and assigned with negative. The FN is the number of reviews that belong to positive category and assigned with negative. And the FP is the number of reviews that belong to negative category and assigned with positive.

The figure 2 shows the accuracy of three methods in the training phase and figure 3 shows the accuracy in the test phase. We can observe from these figures that the accuracy of the ARNN and ARCNN are superior to the BRNN method. We take the best results of 10 iterations and show them in the table 2. By comparing the test accuracy of BRNN (80.22%) with ARNN (89.98%), it is found that using the attention mechanism with the BRNN leading to higher accuracy than only using the BRNN

Table 2. Three methods comparing

Method	Stage	Accuracy (%)
BRNN	Training	89.50
	Test	80.22
ARNN	Training	95.55
	Test	89.98
ARCNN	Training	96.64
	Test	90.34

**Figure 2.** The accuracy of the training stage.**Figure 3.** The accuracy of the test stage.

model. The reason is that the BRNN model cannot specifically capture the sentence information and does not take into account the different priorities of each word in the sentence. And the ARNN model with the attention mechanism can automatically assign low weights to irrelevant words, and high weights are assigned to the relevant words that express emotions. In addition, because of the advantage of CNN for the local information processing, the accuracy of ARCNN (90.34%) is slightly better than ARNN (89.98%).

5. Conclusion

In this paper, we propose a sentiment classification method, named ARCNN, which makes the combination of attention mechanism and BRNN-CNN. In the model, the BRNN can review the information of each time series entered before, and add the attention mechanism to optimize the weight allocation in its hidden layer output. Then enter the convolution neural network to get better features. The design of the model takes into account the calculation of attention weight and the mining of information. The experiments show that the model proposed in this paper can capture the context and assign the context-related weights to each word, so as to effectively excavate the important words and improve the accuracy of classification.

Acknowledgments

This work is supported by the National Key R&D Plan of China under grant no. 2016YFB0800201, the Natural Science Foundation of China under grant no.61070212 and 61572165, the State Key Program of Zhejiang Province Natural Science Foundation of China under grant no. LZ15F020003, the Key research and development plan project of Zhejiang Province under grant no. 2017C01065, the Key Lab of Information Network Security, Ministry of Public Security, under grant no C16603.

6. References

- [1] Murphey Y L, Masrur M A, Chen Z, et al. Model-based fault diagnosis in electric drives using machine learning [J]. *IEEE/ASME Transactions on Mechatronics*, 2006, 11(3):290~303.
- [2] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[C]. *Advances in neural information processing systems*. 2013: 3111-3119.
- [3] Pennington J, Socher R, Manning C D. Glove: Global Vectors for Word Representation [C]. *EMNLP*. 2014, 14: 1532~1543.
- [4] Go A, Bhayani R, Huang L. Twitter sentiment classification using distant supervision [J]. *CS224N Project Report, Stanford*, 2009, 1(12).
- [5] Pang B, Lee L, Vaithyanathan S. Thumbs up?: sentiment classification using machine learning techniques [C]. *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 2002: 79~86.
- [6] Hu M, Liu B. Opinion Feature Extraction Using Class Sequential Rules [C]. *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*. 2006: 61~66.
- [7] Kim Y. Convolutional neural networks for sentence classification [J]. *arXiv preprint arXiv:1408.5882*, 2014.
- [8] Liang and Y. Cai, Polarity Shifting and LSTM Based Recursive Networks for Sentiment Analysis [J]. *Journal of Chinese Information Processing*, 2015. 29(5): p. 152~159.
- [9] Denil M, Bazzani L, Larochelle H, et al. Learning where to attend with deep architectures for image tracking [J]. *Neural Computation*, 2011, 24(8):2151~2184.
- [10] Clarke D. A context-theoretic framework for compositionality in distributional semantics [J]. *Computational Linguistics*, 2012, 38(1): 41~71.
- [11] Irsoy O, Cardie C. Opinion Mining with Deep Recurrent Neural Networks[C]. *Conference on Empirical Methods in Natural Language Processing*. 2014:720~728.
- [12] Serban I V, Sordoni A, Bengio Y, et al. Building end-to-end dialogue systems using generative hierarchical neural network models[C]. *Thirtieth AAAI Conference on Artificial Intelligence*. 2016.
- [13] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks [C]. *Advances in neural information processing systems*. 2014: 3104~3112.
- [14] Wen Y, Zhang W, Luo R, et al. Learning text representation using recurrent convolutional neural network with highway layers [J]. *arXiv preprint arXiv:1606.06905*, 2016.
- [15] Denil M, Bazzani L, Larochelle H, et al. Learning where to attend with deep architectures for image tracking [J]. *Neural Computation*, 2011, 24(8):2151~2184.
- [16] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [J]. *arXiv preprint arXiv:1409.0473*, 2014.
- [17] Xu K, Ba J, Kiros R, et al. Show, attend and tell: Neural image caption generation with visual attention [C]. *International Conference on Machine Learning*. 2015: 2048~2057.
- [18] Sharma S, Kiros R, Salakhutdinov R. Action recognition using visual attention [J]. *arXiv preprint arXiv:1511.04119*, 2015.
- [19] Pollastri G, Mclysaght A. Porter: a new, accurate server for protein secondary structure prediction [J]. *Bioinformatics*, 2004, 21(8): 1719~1720.
- [20] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014