Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Attention-Based LSTM for Target-Dependent Sentiment Classification

Min Yang,[1] **Wenting Tu,**[1] **Jingxuan Wang,**[1] **Fei Xu,**[*2] **Xiaojun Chen**[3]
[1]University of Hong Kong, Hong Kong
[2]Chinese Academy of Sciences, Beijing, China
[3]Shenzhen University, Shenzhen, China
{myang,wttu}@cs.hku.hk jingxuan@hku.hk xufei@iie.ac.cn xjchen@szu.edu.cn

Abstract

We present an attention-based bidirectional LSTM approach to improve the target-dependent sentiment classification. Our method learns the alignment between the target entities and the most distinguishing features. We conduct extensive experiments on a real-life dataset. The experimental results show that our model achieves state-of-the-art results.

Introduction

in real world, people may mention several target entities (or different aspects of a target) in one sentence. Only considering the overall sentiment of the sentence fails to capture the sentiments of different target entities. In this paper, we mainly focus on the task of predicting the sentiment polarity (e.g. positive, negative, neutral) of sentences towards specific targets.

Inspired by the recent success of attention neural network (see Section 2), we present an Attention-based Bidirectional LSTM (AB-LSTM) approach to improve the target-dependent sentiment classification. Our model is a recurrent neural network which includes a bidirectional LSTM network with an additional attention layer on top. It learns to assign attention scores to different word locations according to their intent importance. We then compress the input sequence into a fixed-length vector by computing a weighted sum of the hidden states at each word according to their attention scores. The weights are adaptive to the content of each time step, which makes it possible to assign large weights to the "distinguishing" words. On the other hand, unlike the models which only use the last hidden state (or mean pooling), the attention based model has no difficulty modeling long sequence since it considers different word locations in a relatively even manner. This allows the model to cope with the situation when the input sentence is long and the target string is far from the most distinguishing features. The experiment results show that our model achieves state-of-the-art results.

*Corresponding author

Related work

In order to capture the sentiments on target entities, there are a variety of approaches being proposed for target-dependent sentiment classification (Hu and Liu 2004). Since feature engineering is labor intensive, several recent studies (Dong et al. 2014; Tang et al. 2015) use deep neural networks to tackle this problem. Our model is inspired by the recent success of attention-based neural network (Bahdanau, Cho, and Bengio 2014). To the best of our knowledge, our work is the first one that explores attention-based architecture for target-dependent sentiment classification.

Model

We propose two Attention-based Bidirectional LSTM (AB-LSTM) approaches to improve the performance of target-dependent sentiment classification. AB-LSTM learns to assign attention scores to different word locations according to their relevance to the task. We compress the input sequence into a fixed-length vector by computing a weighted sum of the hidden states of each word according to their attention scores.

AB-LSTM1 Our first attention method compute a weight for every word in the sequence as the dot product of h_t and h_{target}. Here, h_t is the hidden states of time step $t$obtained by bidirectional LSTM based on the current input x_t, h_{target} is the hidden states of the target string. Mathematically, the final sentence representation takes the following form:

$$\mathbf{o} = \sum_{t=1}^{T} a_t h_t, \text{ with } a_t = \frac{\exp(s(h_t \cdot h_{target}; \theta))}{\sum_{t=1}^{T_{target}} \exp(s(h_t \cdot h_{target}; \theta))} \quad (1)$$

where T is the length of the sequence, $s(h_t \cdot h_{target}; \theta)$ is the attention network (θ is the parameter set of attention network), which maps a vector to a real valued score. To form a proper probability distribution over the context, we normalize the scores across the context by using *softmax* function and get attention score a_t. The attention score a_t indicates the importance of the corresponding time step t. In this way, with a properly trained attention network, it assigns higher attention scores to words that are more relevant to the certain task. In our model, $s(h_t \cdot h_{target}; \theta)$ is implemented as another neural network, whose parameters are jointly trained with those of the B-LSTM.

AB-LSTM2 Inspired by (Chen, Bolton, and Manning 2016), our second method computes the attention scores by using a biliner term, instead of the dot product of h_t and h_{target}. With a bilinear term, we can compute the similarity between h_t and h_{target} more flexibly than with a dot product. The output vector of the input sequence is

$$\mathbf{o} = \sum_{t=1}^{T} a_t h_t, \text{ with } a_t = \text{softmax}(h_t^T W_b h_{target}) \quad (2)$$

where $W_b \in \mathbb{R}^{h \times h}$ is used in a bilinear term.

We feed the outpur vector $\mathbf{o}$ to a $softmax$ classifier to predict the sentiment distribution of the target string:

$$\hat{y} = \text{softmax}(V^T \mathbf{o} + b) \quad (3)$$

Here, V and b are parameters to be learned. We train the entire model by minimizing the cross-entropy between the predicted distribution $\hat{y}$ and the ground truth distribution y. The L_2-regularization penalty is used. Given a training sample x^i, its true label $\mathrm{y}^i \in \{1, 2, \cdots, k\}$ where k is the number of categories, and the predicted probabilities $\hat{y}^i \in [0, 1]$, for each label$j \in 1, 2, \cdots, k$, the error is defined as:

$$L = -\sum_{j=1}^{k} \mathrm{I}\{\mathrm{y}^i = j\} \log(\hat{y}^i) + \sum_{\beta \in \Theta} \lambda ||\beta||_2 \quad (4)$$

where Θ represent the parameters of the model. $\mathrm{I}\{\cdot\}$ is an indicator such that $\mathrm{I}\{\text{true}\} = 1$ and $\mathrm{I}\{\text{false}\} = 0$. We use a minibatch stochastic gradient descent (SGD) algorithm together to train the model.

Experiments

Datasets

Twitter conversation (Twitter): The original dataset[1] is a collection of tweets from Twitter by (Dong et al. 2014). The training data consists of 6,248 tweets, and the testing data has 692 tweets. The percentages of positive, negative and neutral tweets in the training and test sets are both 25%, 25%, 50%, respectively.

For the dataset, data preprocessing is performed. We remove non-alphabet characters, numbers, pronouns, punctuation and stop words from the text.

Baseline methods

We compare our approach with several baseline methods, including SVM (Pang, Lee, and Vaithyanathan 2002), SVM with target-dependent features (SVM-dep) (Jiang et al. 2011), AdaRNN-comb (Dong et al. 2014), TC-LSTM (Tang et al. 2015).

Implementation Details

In our work, we use GloVe vectors[2] with 100 dimensions to initialize the word embeddings. Both forward and backward LSTMs have 500 units each and the size of the *softmax* hidden layer in the deep output is 500. We use a minibatch SGD algorithm to train the model. Each minibatch consists 20 input samples. We initialize the recurrent weight matrices as random orthogonal matrices. All the bias vectors are initialized to zero. Any other weight matrices are initialized by sampling from Gaussian distribution with mean 0 and variance 0.01^2. The hyperparameter $\lambda = 0.8$.

[1] http://goo.gl/5Enpu7

[2] http://nlp.stanford.edu/projects/glove

Method	Accuracy (%)	F_1-socre (%)
SVM	62.7	60.2
SVM-dep	63.4	63.3
AdaRNN-comb	66.3	65.9
TC-LSTM	71.5	69.5
AB-LSTM1	71.6	71.2
AB-LSTM2	72.6	72.2

Table 1: Evaluation results

Experimental results

In our experiments, the results are evaluated using classification accuracy and F_1-score. We summarize the experiment results in Table 1. Compared to previous methods, our approaches achieve better results on the experimental dataset. For example, the F_1-score of AB-LSTM2 is 2.7% higher than the state-of-the-art result (Target-dep$^+$). For accuracy, both of our models perform better than Target-dep$^+$and TC-LSTM. This verifies the effectiveness of the proposed approaches.

Conclusion and Future Work

In this paper, we proposed two attention-based models to improve the performance of target-dependent sentiment classification. Experiments on real-lift data showed that our method achieves better or comparable results.

References

Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Chen, D.; Bolton, J.; and Manning, C. D. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *ACL*.

Dong, L.; Wei, F.; Tan, C.; Tang, D.; Zhou, M.; and Xu, K. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *ACL*, 49–54.

Hu, M., and Liu, B. 2004. Mining and summarizing customer reviews. In *ACM SIGKDD*, 168–177.

Jiang, L.; Yu, M.; Zhou, M.; Liu, X.; and Zhao, T. 2011. Target-dependent twitter sentiment classification. In *ACL*, 151–160.

Pang, B.; Lee, L.; and Vaithyanathan, S. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *ACL*, 79–86.

Tang, D.; Qin, B.; Feng, X.; and Liu, T. 2015. Target-dependent sentiment classification with long short term memory. *arXiv preprint arXiv:1512.01100*.