Joint Learning for Targeted Sentiment Analysis

Dehong Ma[†]**, Sujian Li**[†] **and Houfeng Wang**[†‡]
[†]MOE Key Lab of Computational Linguistics, Peking University, Beijing, 100871, China
[‡]Collaborative Innovation Center for Language Ability, Xuzhou, Jiangsu, 221009, China
{madehong,lisujian,wanghf}@pku.edu.cn

Abstract

Targeted sentiment analysis (TSA) aims at extracting targets and classifying their sentiment classes. Previous works only exploit word embeddings as features and do not explore more potentials of neural networks when jointly learning the two tasks. In this paper, we carefully design the hierarchical multi-layer bidirectional gated recurrent units (HMBi-GRU) model to learn abstract features for both tasks, and we propose a HMBi-GRU based joint model which allows the target label of word to have influence on its sentiment label. Experimental results on two datasets show that our joint learning model can outperform other baselines and demonstrate the effectiveness of HMBi-GRU in learning abstract features.

1 Introduction

Targeted sentiment analysis (TSA) aims to extract targets in a text and simultaneously predict their sentiment classes (Hu and Liu, 2004; Jin et al., 2009; Li et al., 2010; Yang and Cardie, 2013). For example, given a sentence *"ESPN poll says Michael Jordan is the greatest basketball athlete"*, the targets are *ESPN* and *Michael Jordan* and their sentiment classes are *Neutral* and *Positive* respectively.

Targeted sentiment analysis can be seen as two tasks: target extraction and sentiment classification. Some researchers have tackled two tasks separately, e.g., target extraction (Liu et al., 2013; Wang et al., 2016a; Yin et al., 2016) and sentiment classification (Tang et al., 2016; Wang et al., 2016b; Ruder et al., 2016). Recently, some researches have attempted to conduct the two tasks jointly and generally see them as sequence labeling problems, where the *B/I/O* labels indicate target boundaries and the *Positive/Neutral/Negative* labels denote sentiment classes (Klinger and Cimiano, 2013; Yang and Cardie, 2013). Mitchell et al. (2013) explore labeling targets and their sentiment classes simultaneously by using the Conditional Random Fields (CRF) approach with traditional manual discrete features, and present three models: pipeline, joint and collapsed, according to different labeling processes of the two tasks. They find that the pipeline method outperforms the joint model on tweet dataset. Further, Zhang et al. (2015) introduce word embedding representations into the CRF framework and find that it is beneficial to integrate word embeddings into handcraft features in TSA regardless of pipeline, joint or collapsed methods.

With the success of deep learning techniques, neural networks have demonstrated their capability of sequence labeling (Collobert et al., 2011; Pei et al., 2014; Chen et al., 2015). However, Zhang et al. (2015) only use word embeddings to enrich features without taking full advantages of neural networks' potential in automatically capturing important sequence labeling features like long distance dependencies and character-level features.

To make better use of neural networks to explore appropriate character-level features and high-level semantic features for the two tasks, we design a hierarchical multi-layer bidirectional gated recurrent units networks (HMBi-GRU) which uses a multi-layer Bi-GRU to automatically learn character features (e.g. capitalization, noun suffix, etc) on letter sequence and model long distance dependencies between words on the concatenation of word embedding and its character features. The learned character features can also address out-of-vocabulary word problems.

In above example, the target label and sentiment label for *Michael Jordon* are "B-Person, I-Person" and "B-Positive, I-Positive", we can see that the boundary information (B, I) of target label and sentiment label is consistent. From the

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4737–4742
Brussels, Belgium, October 31 - November 4, 2018.

view of human, we should first predict the target label and give corresponding sentiment label afterwards. Therefore, we introduce target label information into predicting sentiment label. In this way, our model can know about the target boundary information when predicting the sentiment label. Meanwhile, we also introduce transition matrix (Collobert et al., 2011) to model the dependencies between labels.

We conduct experiments on two datasets, and the performances show that our models outperform other baselines. This verifies the effectiveness of neural networks in TSA. In the experiments, we find that the target label information is important for predicting sentiment label. We also analyze the performance of multi-layer Bi-GRU and hierarchical architecture in learning character features and dependencies between words.

2 Model

We will detailedly introduce our model in this section, and our model is shown in Figure 1. Supposing that a sentence is composed of n words $[w_1, w_2, ..., w_n]$. For each word w_i consists of l_i characters $[c_1, c_2, ..., c_{l_i}]$ and l_i is the length of w_i. We embed all words and characters into low-dimensional real-value vectors which can be learned by language model (Bengio et al., 2003; Mikolov et al., 2013). We represent sentence as a matrix of word embeddings $W = [E_1, E_2, ..., E_n] \in R^{n\times d_w}$. Similarly, word w_i is denoted as a matrix of character embeddings $C_i \in R^{l_i \times d_c}$, and d_w and d_c are the size of word embedding and character embedding respectively.

First, we design a hierarchical two-layer architecture where each layer includes a multi-layer bidirectional Gated Recurrent Units (MBi-GRU). GRU is good at modeling a sequence with the benefits of avoiding the gradient vanishing and exploding problems. For a MBi-GRU, supposing that it has M layers of Bi-GRU, the hidden state on layer $m \in \{1, 2, ..., m\}$ at time $t \in \{1, 2, ..., n\}$ is recursively computed by:

$$h_t^m = \text{BiGRU}(h_t^{m-1}, h_{t-1}^m). \tag{1}$$

where the superscript of h denotes the corresponding layer of a MBi-GRU, and h^0 means the original inputs. BiGRU is bidirectional GRU which is defined as:

$$\text{BiGRU}(x_t, h_{t-1}) = \overrightarrow{h_t} \oplus \overleftarrow{h_t}; \tag{2}$$

$$\overrightarrow{h_t} = \text{GRU}(x_t, \overrightarrow{h_{t-1}}); \tag{3}$$

$$\overleftarrow{h_t} = \text{GRU}(x_t, \overleftarrow{h_{t-1}}). \tag{4}$$

where x_t is inputs which can be word embeddings or the hidden states of other BiGRU. $\oplus$ indicates the operation of concatenating two vectors.

With the matrix of character embeddings C_i as inputs, we utilize a MBi-GRU to learn character-level abstract features for word w_i based on its character embeddings. Through MBi-GRU, we can obtain the hidden states $[h_1^M, h_2^M, ..., h_{l_i}^M]$ on which a max-pooling operation is applied to output the character-level features $r_i \in R^{2d_c}$ for word w_i. The character features of all words in a sentence form a new matrix $C \in R^{n\times 2d_c}$. Next, We concatenate C with the matrix of word embeddings W and denote the concatenation as $F \in R^{n\times(d_w+2d_c)}$. With F as input, We utilize another MBi-GRU to learn the hidden states $H = [h'^M_1, h'^M_2, ..., h'^M_n]$ as the final representations of the sentence. Therefore, the hierarchical two-layer MBi-GRU architecture can learn high-level abstract features with consideration of both character-level and word-level information.

After learning the final representations for sentence, we first project the features: $tf_i = h'^M_i$ of each word into target label space by:

$$y_t^i = f(tf_i \cdot W_p^t + b_p^t) \tag{5}$$

where W_p^t and b_p^t are weight matrix and bias.

As we know, the boundary of a target should be the same as that of its sentiment in sequence label. As the example in Section 1, the target label and sentiment label of *Michael Jordan* are "**B**-Person, **I**-Person" and "**B**-Positive, **I**-Positive" respectively. To learn this kind of consistency, we introduce the target label information into predicting sentiment label by:

$$y_s^i = f(sf_i \cdot W_p^s + b_p^s) \tag{6}$$

where $sf_i = h'^M_i \oplus y_t^i$, W_s^t and b_s^t are weight matrix and bias respectively. This makes our model know the target label information when predicting their sentiment.

For sequence labeling, there usually exist dependencies between labels. Take the target labeling task for example, label *I* will never follow label

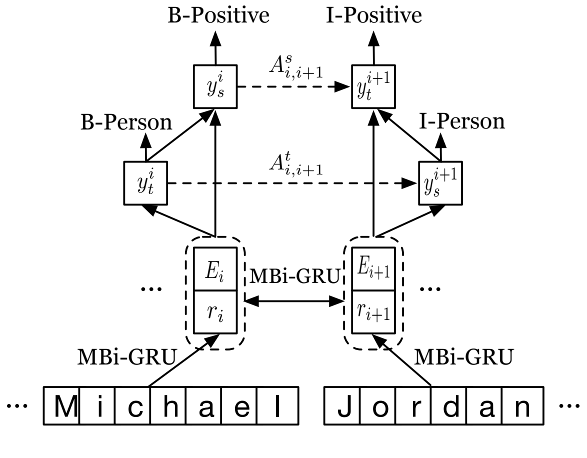

Figure 1: The overall architecture of our model.

O. To consider the influence of label dependencies, we introduce the transition matrix $A_{i,j}$ proposed by Collobert et al. (2011) which measures the probability of jumping from label i to label j.

Given the sentence $x = [w_1, w_2, ..., w_n]$ and the scores $y_t = [y^1_t, y^2_t, ..., y^n_t]$ and $y_s = [y^1_s, y^2_s, ..., y^n_s]$ computed by Eq. 5 and Eq. 6, we get the target labeling scores by summing up transition scores and the scores y^i_t:

$$s(y_t, x, \theta_t) = \sum\nolimits_{i=1}^{n} (A^t_{i-1,i} + y^i_t); \tag{7}$$

where A^t is label transition matrix for target labeling. $\theta_t = \theta \cup \{A^t_{i,j}\}$, and θ denotes parameters of HMBi-GRUs.

Next, we normalize the target label scores over all possible labeling paths of target (i.e., Y_t) by a softmax function:

$$p_t(y_t|x) = \frac{e^{s(y_t,x,\theta_t)}}{\sum_{\hat{y}_t \in Y_t} e^{s(\hat{y}_t,x,\theta_t)}}; \tag{8}$$

We can also use Eq. 7 and Eq. 8 to get the normalized sentiment label scores $p_s(y_s|x)$. To train our model, we define the loss function by:

$$\text{loss} = -\log(p_t(y_t|x)) - \log(p_s(y_s|x)). \tag{9}$$

Finally, we obtain targets label sequence y^*_t and their sentiment label sequence y^*_s which have maximal score $y^*_t = \arg\max_{\hat{y} \in Y_t}(s(x, \hat{y}, \theta_t))$ $y^*_s = \arg\max_{\hat{y} \in Y_s}(s(x, \hat{y}, \theta_s))$. y^*_t and y^*_s can be computed by *Viterbi* algorithm.

3 Experiments

3.1 Setup

To validate the effectiveness of our model, we conduct experiments on two datasets, consisting of English tweets and Spanish tweets, which are constructed by Mitchell et al. (2013)[1]. Table 2 depicts the statistics of data, which contains sentence number, target number and the number of positive target, negative target and neutral target. To evaluate the system performance, we adopt *Precision*, *Recall* and *F-measure*. In our experiments, we evaluate the performance of detecting targets (DT) and targeted sentiment analysis (TSA) which a target is taken as correct only when the boundary and the sentiment are both correctly recognized. We also adopt *Precision*, *Recall* and *F-measure* used in Zhang et al. (2015) to evaluate our model. The reason why we don't compare with Mitchell et al. (2013) is that they only evaluate the beginning of targets along with the sentiment expressed towards it.

Datasets	#Sent	#Target	#Pos	#Neg	#Neu
English	2350	3288	707	275	2306
Spanish	5145	6658	1555	1007	4096

Table 1: Statistics of Datasets.

In our experiments, we use embeddings from Pennington et al. (2014)[2] and Cieliebak et al. (2017)[3] for English words and Spanish words respectively. The character embeddings are initialized by Xavier (Glorot and Bengio, 2010) and their dimension is 50. In our model, all unknown words, weight matrices and biases are initialized by Xavier Glorot and Bengio (2010). The dimensions of the character-level and word-level hidden states in MBi-GRU are set to 300 and 600 respectively. The layer number of multi-layer bidirectional GRU is set to 2. To avoid overfitting, we adopt dropout on embeddings, sf_i and tf_i, and the dropout rate is set to 0.5. The word embeddings and character embeddings will be tuned during training. Finally, we utilize Adam (Kingma and Ba, 2014) to optimize all parameters of our model.

3.2 Baselines

To investigate the performance of our joint model, we compare it with several baselines as follows:

- **Discrete** uses traditional discrete features as

[1] http://www.m-mitchell.com/code/index.html

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

[3] https://spinningbytes.com/resources/embeddings/

Model	English						Spanish					
	DT			TSA			DT			TSA		
	P	R	F	P	R	F	P	R	F	P	R	F
Discrete	59.55	34.06	43.30	43.09	24.67	31.35	71.08	47.56	56.96	46.36	31.02	37.15
Neural	54.45	42.12	47.17	37.55	28.95	32.45	65.05	47.79	55.07	40.28	29.58	34.09
Integrate	**61.47**	49.28	54.59	44.62	35.84	39.67	**71.32**	61.11	65.74	46.67	39.99	43.02
Bi-GRU	58.13	43.46	49.62	45.76	32.29	37.73	65.24	53.02	58.45	46.33	37.50	41.45
MBi-GRU	58.27	49.01	53.24	45.80	35.21	39.81	66.14	60.07	62.95	45.61	40.04	42.64
HBi-GRU	57.24	**53.88**	55.41	44.94	38.60	41.52	68.24	61.81	64.82	46.53	42.21	44.18
No-Target	61.24	52.44	56.39	45.90	39.21	42.21	66.72	63.57	65.10	45.06	43.31	44.17
OURS	60.12	53.68	**56.98**	**46.52**	**39.99**	**42.87**	68.64	**63.66**	**66.01**	**48.09**	**43.44**	**45.61**

Table 2: Performance comparison of our models with the baselines.

inputs and multi-label CRF which contains two separate output clique potentials and two separate edge clique potentials for target extraction and sentiment classification respectively. There also exist links between target labels and sentiment labels for each word (Zhang et al., 2015).

• **Neural** uses word embeddings transformed with non-linear function as inputs, and others are the same as *Discrete* model (Zhang et al., 2015).

• **Integrated** integrates both discrete features and word embeddings into the same CRF framework and other settings are the same as *Discrete* (Zhang et al., 2015).

• **Bi-GRU** only uses word embeddings as inputs, and Bi-GRU is employed to learn representations for sentence.

• **MBi-GRU** also uses word embeddings as inputs, but MBi-GRU is utilized to model sentence.

• **HBi-GRU** first uses Bi-GRU to learn character level features for each word. Then, character level features and word embeddings are concatenated as inputs for another Bi-GRU to learn final representations for sentence.

• **No-Target** uses HMBi-GRU to learn representations for sentence, but h'^{M}_{i} (depicted in Section 2) are used to predict target label and sentiment label separately. *No-Target* doesn't let target label information to affect sentiment label. This is the biggest difference between *No-Target* and ours.

It is noticed that all of *Bi-GRU*, *MBi-GRU* and *HBi-GRU* use transition matrix to model the dependencies between labels and introduce target label information into predicting sentiment label.

3.3 Analysis

Table 2 displays the performance comparison of our models with the baselines. We can see that *Discrete* gets the worst results on English dataset, and *Neural* gets the worst results on Spanish dataset. The *Integrate* greatly improves the performances on both datasets because discrete features and word embeddings can complement each other.

Bi-GRU greatly improves the performance compared with *Discrete* and *Neural* but gets worse performance than *Integrate*. This verifies the effectiveness of neural networks in TAS. However, simple neural networks are not enough to acquire better results. *MBi-GRU* learns high-level features via multi-layer bidirectional GRU and achieves comparable results compared with *Integrate*.

Nevertheless, *Bi-GRU* and *MBi-GRU* do not make full use of character-level features. *HBi-GRU* incorporates character-level features by Bi-GRU on letter sequence of word. We can see that *HBi-GRU* improves about 1.85% and 1.16% in TSA on both datasets compared with *Integrate*. The performance of *HBi-GRU* demonstrates the importance of character-level features in TSA, and the hierarchical architecture is good at leaning multi-level (character-level, word-level) features.

Our model improves 3.20%, 2.59% in TSA and 2.39%, 0.27% in DT on both datasets compared with the existing best system: *Integrate*. Compared with *No-Target*, our model introduces target label information into predicting sentiment label and improves about 0.66%, 1.44% in TSA and 0.59%, 0.91% in DT on both datasets. The improvements demonstrate that target label information plays important roles in predicting sentiment label. It is noticed that the results of our model in

DT are also improved compared with *No-Target*. The reason may be that the gradients from sentiment loss have positive effects on detecting targets.

In a word, our model achieves state-of-the-art in DT and TSA on both datasets. Character-level features play great roles in DT and TSA, and HMBi-GRU is good at learning multi-level features. It is useful to learn boundary consistence by introducing target label information into predicting sentiment label.

3.4 Case Study

Here, we use a tweet from English Dataset as a case study, and the tweet is "Congratulations to our Champ Roger Federer ...". We apply *No-Target* and our model on the tweet. *No-Target* and our model get the same target labels: [O,O,O,O,B-Person,I-Person,...], and we can see that both models correctly extract the target: *Roger Federer*, and this results show the effectiveness of both models in detecting targets. Our model successfully obtains the correct sentiment labels: [O,O,O,O,B-Positive,I-Positive,...]. However, *No-Target* predicts a wrong sentiment label sequence: [O,O,O,B-Positive,I-Positive,O,...]. We can see that *No-Target* wrongly regard *Champ* as the beginning position and ignore *Federer*. The reasons are that the first letter of Champ is capitalized, which may mislead No-Target and there is no correlation between target and sentiment label. In our model, we incorporate target label information into predicting sentiment label. Therefore, our model tends to force target and sentiment label to have same boundary information.

This case study shows that the target label information plays important roles in predicting sentiment label because they share the same boundary information.

4 Related Work

Early works on target sentiment analysis were based on subjects and features. For example, Yi et al. (2003) extracted all references to the given subject and determined the sentiment of each reference. Hu and Liu (2004) first proposed several techniques to mine the product features that customers have expressed their opinions and determined their sentiment, and Popescu and Etzioni (2007) utilized unsupervised methods to identify opinions with respect to features and determine the polarity of opinions. Jin et al. (2009) proposed a novel lexicalized HMMs model to mine customer reviews of a product and extract highly specific product related entities which reviewers expressed their opinion, and they also identified the sentiment of opinion entities. The works of (Yang and Cardie, 2013) and (Li et al., 2010) are similar to (Jin et al., 2009). However, these works only take pre-defined features into account and can not find new features. To automatically extract targets and predict their sentiment, Mitchell et al. (2013) first proposed a conditional random fields (CRF) framework to jointly detect entities and identify their sentiment. Based on the work of (Mitchell et al., 2013), Zhang et al. (2015) explored the effect of word embeddings and automatic feature combinations by extending a CRF baseline using neural networks.

We propose a neural networks based joint model which extracts targets and their sentiments simultaneously. Our model takes full advantages of neural networks' potential in capturing sequence labeling features such as long distance dependencies and character-level features. Furthermore, Our model allows the target label to have positive effects on their sentiment label because target label shares boundary information with sentiment label.

5 Conclusion

In this paper, we propose a HMBi-GRU based joint model for targeted sentiment analysis. Our model will simultaneously extract targets and predict their sentiment. Furthermore, our model introduces target information into predicting corresponding sentiment label. Experiments show that the well-designed neural networks can greatly improve the result for targeted sentiment analysis, and target label information plays great roles in predicting sentiment label.

Acknowledgments

We thank anonymous reviewers for their insightful suggestions. Our work is supported by National Natural Science Foundation of China under Grant No.61433015 and the National Key Research and Development Program of China under Grant No.2017YFB1002101. The corresponding author of this paper is Houfeng Wang.

References

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *JMLR*, 3(Feb):1137–1155.

Xinchi Chen, Xipeng Qiu, Chenxi Zhu, Pengfei Liu, and Xuanjing Huang. 2015. Long short-term memory neural networks for chinese word segmentation. In *EMNLP*, pages 1197–1206.

Mark Cieliebak, Jan Deriu, Dominic Egger, and Fatih Uzdilli. 2017. A twitter corpus and benchmark resources for german sentiment analysis. *SocialNLP*, page 45.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *JMLR*, 12(Aug):2493–2537.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, pages 249–256.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *SIGKDD*, pages 168–177. ACM.

Wei Jin, Hung Hay Ho, and Rohini K Srihari. 2009. A novel lexicalized hmm-based learning framework for web opinion mining. In *ICML*, pages 465–472.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Roman Klinger and Philipp Cimiano. 2013. Bi-directional inter-dependencies of subjective expressions and targets and their value for a joint model. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 848–854.

Fangtao Li, Chao Han, Minlie Huang, Xiaoyan Zhu, Ying-Ju Xia, Shu Zhang, and Hao Yu. 2010. Structure-aware review mining and summarization. In *ACL*, pages 653–661.

Kang Liu, Heng Li Xu, Yang Liu, and Jun Zhao. 2013. Opinion target extraction using partially-supervised word alignment model. In *IJCAI*, volume 13, pages 2134–2140.

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

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *ENMLP*, pages 1643–1654.

Wenzhe Pei, Tao Ge, and Baobao Chang. 2014. Max-margin tensor neural network for chinese word segmentation. In *ACL*, pages 293–303.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

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.

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016. A hierarchical model of reviews for aspect-based sentiment analysis. *arXiv preprint arXiv:1609.02745*.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective lstms for target-dependent sentiment classification. In *International Conference on Computational Linguistics*, pages 3298–3307.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016a. Recursive neural conditional random fields for aspect-based sentiment analysis. *arXiv preprint arXiv:1603.06679*.

Yequan Wang, Minlie Huang, Li Zhao, and Xiaoyan Zhu. 2016b. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the conference on empirical methods in natural language processing*, pages 606–615.

Bishan Yang and Claire Cardie. 2013. Joint inference for fine-grained opinion extraction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1640–1649.

Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack. 2003. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 427–434. IEEE.

Yichun Yin, Furu Wei, Li Dong, Kaimeng Xu, Ming Zhang, and Ming Zhou. 2016. Unsupervised word and dependency path embeddings for aspect term extraction. *arXiv preprint arXiv:1605.07843*.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2015. Neural networks for open domain targeted sentiment. In *EMNLP*, pages 612–621.