Target-specific Convolutional Bi-directional LSTM Neural Network for Political Ideology Analysis

Xilian Li[1], Wei Chen[1] ⋆, Tengjiao Wang[1], and Weijing Huang[1]

[1] Key Lab of High Confidence Software Technologies (MOE), School of EECS, Peking University, Beijing, 100871, China
{xilianli,pekingchenwei,tjwang,huangweijing}@pku.edu.cn

Abstract. Ideology detection from text plays an important role in identifying the political ideology of politicians who have expressed their beliefs on many issues. Most existing approaches based on bag-of-words features fail to capture semantic information. And other sentence modeling methods are inefficient to extract ideological target context which is significant for identifying the political ideology. In this paper, we propose a target-specific Convolutional and Bi-directional Long Short Term Memory neural network (CB-LSTM) which is suitable in intensifying ideological target-related context and learning semantic representations of the text at the same time. We conduct experiments on two commonly used datasets and a well-designed dataset extracted from tweets. The experimental results show that the proposed method outperforms the state-of-the-art methods.

Keywords: Ideology Detection, Ideological Target, Convolutional neural network, Recurrent neural network

1 Introduction

Recently, more and more politicians turn to social networks such as Twitter and Facebook to express their beliefs instead of relying on traditional interviews or magazines. Ideology detection from these user generated texts play an important role in identifying the political ideology (conservative or liberal) of politicians on many issues, such as predicting poll ratings [8] or evaluating political leadership abilities, and has attracted increasing research interests [1].

Text-based ideology detection remains two significant challenges. One is the semantic information capturing. For example, two phrases "*tax less*" and "*increase taxes* about "*taxation*" from opposite alignments are considered highly similar if ignoring the syntax and word orders. Another crucial challenge is how to detect the target or target-related semantic context which makes it more difficult than sentiment analysis [6]. One sentence "*Abortion is murder of a human being.*" from conservatives opposes "*abortion*", while another sentence "*We can't deprive the rights that a woman can decide what happens with her baby.*" from liberals supports "*abortion*", even it does not contain keyword "*abortion*". Both texts express negative sentiment towards the same

⋆ Corresponding author.

target, but the ideologies are just opposite which proved the importance of recognizing key target.

However, existing studies for ideology detection often fail to capture semantic information and extract ideological target context at the same time. Traditional methods exploit machine learning algorithms [7,10] to build ideology classifiers based on bag-of-words representations with various hand-crafted features [4,9], but neglect the intrinsic contextual relations between sub-sentences. Recently, dominant neural networks which focus on sentence representations can grasp the semantic information while inefficient to extract ideological target context. For instance, some recurrent neural networks [11,24] are capable of capturing abundant semantic correlations for sentiment analysis [3], but are not dedicated for extracting local features. Other frameworks represented by recursive neural networks fetch lexical characteristics to identify ideological influence [13], while depend on complex syntactic tree parsing [15,22]. Convolutional models [14] which toward stance classification [23] can extract local features, yet can not learn sequential correlations. These methods alone are insufficient for purifying target-related context or learning contextual information to the greatest extent.

To address these limitations, we propose a target-specific Convolutional Bi-directional LSTM neural network (CB-LSTM) which is suitable in capturing target-related context and learning semantic representations simultaneously. We apply a modified convolutional structure which is designed to obtain target-related context and reserve position information between n-grams. Then we integrate the output of convolutional layer into BLSTM, which helps to avoid missing long text information, to learn the sequential correlations. By combining the convolutional and LSTM structure, our model takes advantage of both convolutional neural models and recurrent neural networks.

We evaluate the performance on Convote, IBC [13] and a well-refined Twitter dataset with the work of our team [5]. Experimental results show that the CB-LSTM model has superior performances over other existing methods. The main contributions of this work can be summarized as follows:

- We present a unified neural network CB-LSTM which can extract core local features and encode sequential correlations between sub-sentences.
- We adopt the CB-LSTM model to enhance target-related context and learn semantic representations simultaneously for ideology detection task. The empirical results show that it outperforms other state-of-the-art approaches.
- We introduce an innovative and generic method to obtain ideology-related dataset from massive data.

2 CB-LSTM Model

We propose a target-specific convolutional bi-directional long-short term neural network to capture the target context and semantics of the text which are further used as features for ideology detection. First the representations of target semantic or n-grams for each sentence are learned through convolutional structure in Fig.1 (a) with word embeddings as inputs. Next, the target-level vectors are concatenated with the word embeddings and are fed into the BLSTM network. Finally, the output is passed to a

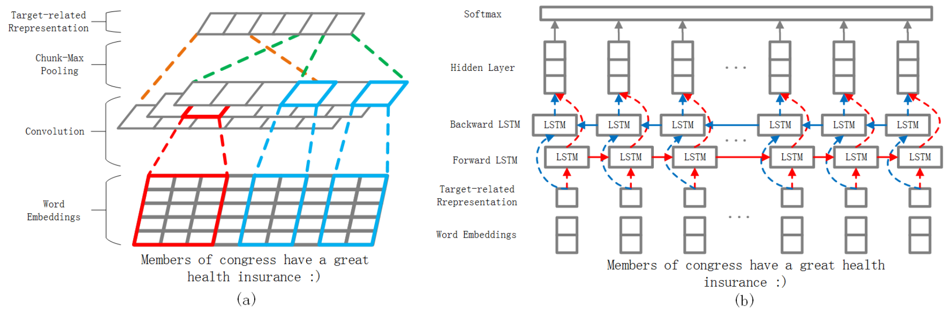

Fig. 1. (a) is the CNN architecture for learning target representations. (b) is the main structure of BLSTM. Dashed arrows indicate dropout layers are applied.

hidden layer and the softmax layer determines the ideology score with maximum probability of the word as shown in Fig.1 (b).

2.1 Target Context Representation through CNN

In this section, we will present a convolutional network with multiple filters to learn semantic representations of the implied target n-grams. In our network, the CNN only has one layer of convolution on top of pre-trained word vectors. In addition, there is no fully connected layer at the end.

A convolution operation involves multiple filters. They are applied to a window of h words in a tweet t expressed as $x_{i:i+h-1}$ which is the concatenation of words $w_1, w_2, ..., w_n$. As a result, some feature maps are produced. A local feature $t_i \in \mathbb{R}^{1\times(|t|-h+1)}$ related to the ideological target is computed by

$$t_i = f(W_t \cdot x_{i:i+h-1} + b_t) \tag{1}$$

where $W_t \in \mathbb{R}^{h_f \times h \cdot k}$ is the weight matrix between input and convolution, b_t is a bias term for each feature map f is a hyperbolic tangent($tanh$) function and k is the dimension of word vectors. We use three filters whose window sizes are 2, 3 and 4.

Then a pooling layer to get a fixed length vector is applied. We refer to a Chunk-Max Pooling [18] operation which has been proved improvement to capture necessary position information between target features.

2.2 Sentence Representation with BLSTM

In this section, the target-level representation vectors learned through the convolutional layer are concatenated with the original word embeddings. Then we feed the concatenated vectors with intensified target semantic information into BLSTM to learn rich contextual correlations of the sentence. The detailed structure is shown in Fig.1 (b).

As a special RNN structure, LSTM has shown strong capability for modeling long-range dependencies over standard RNN in various previous studies [12,16]. The key element of LSTM is cell state c_t. It is able to remove or add information to the cellular state through a carefully designed structure called "gates": the *input gate* i_t, *forget gate* f_t and *output gate* o_t, which are used to protect and control cell status and avoid gradient vanishing or exploding [11]. We refer to the structure of Graves [20] and the main steps are as follows.

$$\begin{aligned}
i_t &= f(W_i \cdot [c_{t-1}, h_{t-1}, x_t] + b_i) \\
f_t &= f(W_f \cdot [c_{t-1}, h_{t-1}, x_t] + b_f) \\
c_t &= f_t \circ c_{t-1} + i_t \circ tanh(W_c[h_{t-1}, x_t] + b_c) \\
o_t &= f(W_o \cdot [c_{t-1}, h_{t-1}, x_t] + b_o) \\
h_t &= o_t \circ tanh(c_t)
\end{aligned} \tag{2}$$

where $\circ$ denotes element-wise multiplication, f is the sigmoid function and $b_i, i \in [i, f, c, o]$, is the different bias term belonging to different gates.

BLSTM has both a forward and a backward LSTM in the hidden layer. The forward LSTM captures the past feature context, while the backward LSTM captures the future feature information. Dropout is used to avoid overfitting.

2.3 Ideology Detection

In the last layer of the CB-LSTM, we intend to predict the ideology polarity of one sentence based on learned semantic sentence representation h_o with several hidden layers and a *softmax* function. The output is defined as

$$\widetilde{y_i} = \frac{exp((W_o \cdot h_o + b)^i)}{\sum_{j=1}^{k} exp((W_o \cdot h_o + b)^j)} \tag{3}$$

where k is the number of known labels.

The loss function applied is cross-entropy error between predicted label $\widetilde{y_i}$ and true label y_i.

$$L(X, y) = \sum_{j=1}^{N} \sum_{i=1}^{k} \widetilde{y_i^j} \cdot log(y_i^j) + \lambda ||\theta||^2 \tag{4}$$

where N is the number of train data, λ is the regularization coefficient and θ stands for the model parameters. We apply RMSprop algorithm to update the parameters by minimizing the loss function on training dataset.

3 Experiments

3.1 Datasets

To perform the effectiveness of our model, we focus on Convote [19], IBC [13] and Twitter data. Convote dataset is the Congressional debates data that has annotation on the author level. IBC is the Ideological Books Corpus which contains a million sentences written by authors with well-known political leanings. It was after annotated on the sentence and phrase level by crowdsourcing [13].

Twitter Datasets: In order to get enough ideology-related datasets, we collected 88 liberal and 131 conservative user accounts from American political forums and crawled

Table 1. The overall information of the datasets. #class is the number of classes, #sen represents the number of sentences in per dataset, #word is the vocabulary size of words, #avgLen is the average number of words contained in each sentence.

Dataset	#class	#sen	#word	#avgLen
Convote	2	7,816	15,024	25.5
IBC	2	3,412	13454	41.4
Twt4w1kw	3	40,000	23246	26.5
Twt2w1kw	3	20,000	19958	23.4
Twt2w2kw	3	20,000	19958	23.4

668763 tweets posted by them until June 2015 as our Twitter corpus. Next we apply an innovative transfer learning model for accurate event detection [5] to extract 16 category-level ideology keyword lists from Wikipedia, ranked by Chi-square score which is the degree of prominence used to differentiate from other keywords in the same categories. After correction by political domain experts, we get the final outstanding ideology keyword lists. With these comprehensive keyword lists, we filter the tweets corpus. Table 1 provides detailed information about each final dataset. Suffix $i$kw represents that the dataset is filtered by i keywords.

3.2 Experimental Setting

We preprocess the datasets as follows. For all datasets, we use Standford CoreNLP[1] tool to get tokenization of the sentences. We split each dataset into training and testing sets with 80/20, with 10% of the training set as development set. The evaluation metric of all these datasets is accuracy.

All of the parameters are initialized from a uniform distribution *uniform*(-0.05, 0.05). We pre-train a 100-dimension word embedding matrix on a 2 billion tweets set crawled in the year of 2015 and 2016 by skip-gram model of Google word2vec[2]. Other hyperparameter settings of our neural networks are depending on which dataset is being used. For convolutional layers, we use some of the hyper-parameters following previous work [14]. A mini-batch size is 50. We set the size of memory dimension as 50 and the learning rate of RMSprop to 0.01. Dropout rate is set to be 0.5 in convolutional and BLSTM layer. L2 regularization of le-6 is used to the last softmax layer.

4 Results and Discussion

4.1 Ideology detection

The experimental results are shown in Table 2. We compare the traditional baselines SVM and Logistic Regression to other neural network approaches. The experiment results show that the latter models outperform the formers on all datasets. It proves the deep learning methods are better at capturing semantic information. It's worth noting

[1] `http://stanfordnlp.github.io/CoreNLP/tokenize.html`
[2] `http://code.google.com/p/word2vec/`

Table 2. The accuracy of ideology detection on five datasets. The top block includes the baselines. The models in the second block are the state-of-the-art neural networks. The penultimate is a fast-trained method. The best method in each settings is in bold.

Model	Convote	IBC	Twt4w1kw	Twt2w1kw	Twt2w2kw
SVM + BoW	68.1%	64.5%	62.6%	64.2%	64.4%
LR + BoW	67.7%	65.3%	63.3%	63.7%	64.3%
SVM + word2vec	69.8%	62.7%	65.9%	66.2%	65.5%
LR + word2vec	67.8%	62.8%	66.5%	67.0%	66.7%
RNN1-(w2v)	70.2%	67.1%	61.4%	62.9%	67.5%
RNN2-(w2v)	-	**69.3%**	-	-	-
CNN-nonstatic-rand	69.9%	62.7%	63.4%	65.3%	67.4%
CNN-static-word2vec	71.3%	63.3%	65.1%	67.0%	68.0%
CNN-nonstatic-word2vec	73.5%	66.1%	67.7%	68.9%	69.4%
LSTM	68.1%	64.7%	67.1%	66.4%	65.9%
BLSTM	69.7%	65.5%	67.5%	67.9%	67.4%
CNN-LSTM	73.1%	68.1%	64.1%	65.0%	70.5%
FastText (Mikolov et al. 2016)	71.8%	64%	64.7%	66.8%	67.0%
CB-LSTM	**75.2%**	68.9%	**69.1%**	**69.8%**	**71.8%**

that, as a rapid approach, FastText [17] has better performance far more than the basic methods and also obtains competitive results compared to several neural networks.

When comparing the CB-LSTM to the RecursiveNNs which include RNN1-w2v and RNN2-(w2v) [13] , we find it outperforms the latter on all datasets except IBC. We believe the reason is that RNN2-(w2v) significantly benefit from phrase-level annotations with a syntactic parsing tree. But most of the datasets lack phrase annotations, RNN2-(w2v) is only applied to IBC dataset. Despite the competitive results, our model does not require complicated syntactic parsing process, which means that it is more simple to train with lower complexity.

By comparing many variants of LSTM and CNN [14] implemented by us, we discover that (1) BLSTM has better performance than single directional LSTM. Because the BLSTM is able to learn contextual information at every time step from both previous and future text. (2) CNNs achieve better results than RecursiveNNs on most of the datasets which illustrates that CNNs are good at capturing the context information through the convolution layer and extract key features by pooling layer. (3) CNN-LSTM outperforms LSTM on all datasets which strongly proves that the importance of target-context representations.

With the ability of intensifying target-related semantic context and representing contextual correlations with lower time complexity, the CB-LSTM model outperforms or achieves competitive results with respect to other methods. Furthermore, we find that all methods perform better on Twt2w2kw dataset than Twt2w1kw which reveals that our methods can obtain better results with more accurate data.

4.2 Model Analysis

We do extensive experiments to check how the performance changes with various parameters. Fig.2 shows the stability of CB-LSTM, CNN(CNN-nonstatic-word2vec),

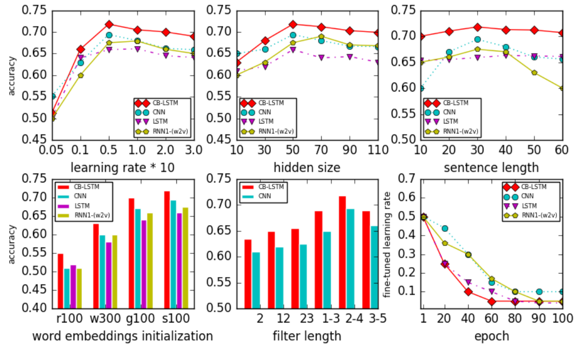

Fig. 2. Accuracy for ideology detection as a function of five hyperparameters: learning rate, hidden size, sentence length, word embeddings initialization and filter length.

LSTM and RNN1-(w2v) for different learning rates, hidden sizes, sentence lengths, word embeddings initialization and filter lengths. In the fourth subgraph, r100 represents random 100-dimension word embeddings. w300 and g100 represent the available Google word2vec and Glove word embeddings[3]. Specifically, s100 stands for our pre-trained word vectors with 100-dimension.

As shown in Fig.2, all models are relatively smooth when learning rate is greater than 0.05. The hidden size and sentence length do not have a significant effect on the accuracy. Moreover, we find that CB-LSTM always performs better than others which proved it is good at capturing contextual information. For word embeddings initialization, the accuracy is higher with Glove initial vectors. In addition, our pre-trained word vectors based on unlabeled tweets perform best. The last but one shows that multiple filter lengths of 2,3,4 performs best among all filter configurations. The last sub-graph shows that the fine-tuned learning rates deviate about 0.45 from initial values.

5 Conclusion

In this paper we proposed a novel, integrated model, CB-LSTM for political ideology detection. Our model is ideal for intensifying ideological target-related context and learning comprehensive contextual information at the same time without depending on complicated tree parsing. The experiment results demonstrate that our model outperforms several baselines on real datasets. Furthermore, we publish an ideology-related dataset which can be used to many ideology analysis tasks. Our model can be also applied into general sentiment analysis, question classification and many other tasks.

In the future, we plan to integrate other public profile information of users, such as marriage status and income, into the current model to better understand users ideology.

Acknowledgments. This research is supported by the Natural Science Foundation of China (Grant No.61572043) and National Key Research and Development Program (Project Number: 2016YFB1000704)

[3] `http://nlp.standford.edu/projects/glove/`

References

1. Conover, M. D., Gonalves, B., Ratkiewicz, J., Flammini, A.: Predicting the political alignment of twitter users. In: IEEE PASSAT and SocialCom, pp. 192-199 (2011).
2. Conover, M., Ratkiewicz, J., Francisco, M. R., Gonalves, B., Menczer, F., Flammini, A.: Political polarization on twitter. In: ICWSM, 133, pp. 89-96 (2011).
3. Sarlan, A., Nadam, C., Basri, S. Twitter sentiment analysis. In: IEEE ICIMU, pp. 212-216 (2014).
4. O'Connor, B., Balasubramanyan, R., Routledge, B.R.: From tweets to polls: Linking text sentiment to public opinion time series. In: ICWSM, 11(122-129), pp.1-2 (2010).
5. Huang, W., Wang, T., Chen, W.: Category-Level Transfer Learning from Knowledge Base to Microblog Stream for Accurate Event Detection. In: DASFAA (2017).
6. Menini, S., Tonelli, S.: Agreement and disagreement: Comparison of points of view in the political domain. In: ACL, pp. 2461-2470 (2016).
7. Pennacchiotti, M. and Popescu, A.M.: Democrats, Republicans and Starbucks afficionados: User classification in Twitter. In: Proc. of KDD, pp. 430-438 (2011).
8. Gruzd, A., Roy, J.: Investigating political polarization on Twitter: A Canadian perspective. In Policy & Internet, 6(1), pp.28-45 (2014).
9. Gerrish, S., Blei, D. M.: Predicting legislative roll calls from text. In: ICML, pp. 489-496 (2011).
10. Volkova, S., Coppersmith, G., Van Durme, B.: Inferring User Political Preferences from Streaming Communications. In: ACL, pp. 186-196 (2014).
11. Hochreiter, S., Schmidhuber, J.: Long short-term memory. In: Neural computation, 9(8), pp. 1735-1780 (1997).
12. LeCun, Y., Bengio, Y. and Hinton, G.: Deep learning. In: Nature, 521(7553), pp. 436-444 (2015).
13. Iyyer, M., Enns, P., Boyd-Graber, J., Resnik, P.: Political ideology detection using recursive neural networks. In: ACL, pp. 1113-1122 (2014).
14. Kim, Y.: Convolutional neural networks for sentence classification. In: EMNLP, pp. 1746-1751 (2014).
15. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent Convolutional Neural Networks for Text Classification. In: AAAI, Vol. 333, pp. 2267-2273 (2015).
16. Tai, K. S., Socher, R., Manning, C. D.: Improved semantic representations from tree-structured long short-term memory networks. In: ACL (2015).
17. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 (2016).
18. Chen, Y., Xu, L., Liu, K., Zeng, D., Zhao, J.: Event extraction via dynamic multi-pooling convolutional neural networks. In: ACL (1), pp. 167-176 (2015).
19. homas, M., Pang, B., Lee, L.: Get out the vote: Determining support or opposition from Congres- sional floor-debate transcripts. In: EMNLP, pp. 327-335 (2006).
20. Graves, A., Jaitly, N., Mohamed, A. R.: Hybrid speech recognition with deep bidirectional lstm. In: ASRU, IEEE Workshop, pp. 273-278 (2013).
21. Huang, E. H., Socher, R., Manning, C. D., Ng, A. Y.: Improving word representations via global context and multiple word prototypes. In: ACL, pp.873-882 (2012).
22. Li, D., Furu, W., Chuanqi, T., Duyu, T.: Adaptive recursive neural network for target-dependent twitter sentiment classification. In: ACL (2), pp. 49-54 (2015).
23. Yu, N., Pan, D., Zhang, M., Fu, G.: Stance Detection in Chinese MicroBlogs with Neural Networks. In ICCPOL of Oriental Languages pp. 893-900 (2016).
24. Augenstein, I., Rocktschel, T., Vlachos, A., Bontcheva, K.: Stance detection with bidirectional conditional encoding. arXiv preprint arXiv:1606.05464 (2016).