NNEMBs at SemEval-2017 Task 4: Neural Twitter Sentiment Classification: a Simple Ensemble Method with Different Embeddings

Yichun Yin
Peking University
yichunyin@pku.edu.cn

Yangqiu Song
HKUST
yqsong@cse.ust.hk

Ming Zhang
Peking University
mzhang_cs@pku.edu.cn

Abstract

Recently, neural twitter sentiment classification has become one of state-of-the-arts, which requires less feature engineering work compared with traditional methods. In this paper, we propose a simple and effective ensemble method to further boost the performances of neural models. We collect several word embedding sets which are publicly released (often are learned on different corpus) or constructed by running Skip-gram on released large-scale corpus. We make an assumption that different word embeddings cover different words and encode different semantic knowledge, thus using them together can improve the generalizations and performances of neural models. In the SemEval 2017, our method ranks 1st in Accuracy, 5th in AverageR. Meanwhile, the additional comparisons demonstrate the superiority of our model over these ones based on only one word embedding set. We release our code [1] for the method replicability.

1 Introduction

Twitter sentiment classification has attracted a lot of attention (Dong et al., 2015; Nakov et al., 2016; Rosenthal et al., 2017), which aims to classify a tweet into three sentiment categories: negative, neutral, and positive. Tweet text has several features: written by the informal language, hash-tags and emoticons indicate sentiments, and sometimes is sarcasm, which make decisions of tweet sentiment hard for machines. With releases of annotated datasets, more researchers prefer to use the twitter sentiment classification as one testbed to evaluate their proposed models.

Traditional methods (Mohammad et al., 2013) for twitter sentiment classification use a variety of hand-crafted features including surface-form, semantic and sentiment lexicons. The performances of these methods often depend on the quality of feature engineering work, and building a state-of-the-art system is difficult for novices. Moreover, these designed features are presented by the one-hot representation which cannot capture the semantic relativeness of different features and proposes a problem of feature sparsity. To address this, Tang et al. (2014) induced sentiment-specific low-dimensional, real-valued embedding features for twitter classification, which encode both semantics and sentiments of words. In the experiments, the embedding features and hand-crafted features obtain similar results, and also they are complementary for each other in the system. With the developments of neural networks in natural language processing, neural sentiment classification (Severyn and Moschitti, 2015; Deriu et al., 2016) has attracted a lot of attention recently and become the state-of-the-arts. These methods first learn word embeddings from large-scale twitter corpus, then tune neural networks by the tweets which have distant labels, and finally fine-tune the proposed models by the annotated datasets.

Learning word embeddings using in-domain data is an effective way to boost model performances (Mikolov et al., 2013; Yin et al., 2016). However, collecting large-scale twitter corpus is often time-consuming. In this paper, we use the different word embedding sets to boost the performances of our neural networks, which only include released different word embeddings sets and the word embedding set derived from the released Yelp large-scale datasets by Skip-gram (Mikolov et al., 2013). A simple and effective ensemble

[1] https://github.com/zwjyyc/NNEMBs

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 621–625,
Vancouver, Canada, August 3 - 4, 2017. ©2017 Association for Computational Linguistics

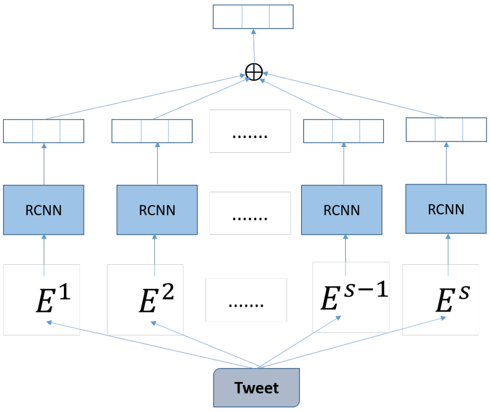


Figure 1: Overview of our method.

method is proposed, which takes different word embedding sets as input to train neural networks and predicts labels of testing tweets by merging all output of neural models. Our ensemble method show its effectiveness in SemEval 2017, though most of used word embedding sets are not learned from twitter corpus, which can be explained that different embedding sets has different vocabularies and encode different parts of sentiment knowledge. Moreover, we conduct additional experiments to analyze our model.

2 Method

In this section, we describe the details of our method, which is illustrated in Figure 1. We feed different word embedding sets into neural networks and train these neural networks separately. When predicting the labels of tweets in testing set, we sum label probabilities of all neural network to make final decisions.

2.1 Neural Network

We have many choices of neural networks (e.g., LSTM, RNN and GRU) for our method, here we consider RCNN (Lei et al., 2016) in our method. RCNN has non-consecutive convolution and adaptive gated decay, which aims to capture longer-range, non-consecutive patterns in a weighted manner.

Given a sequence of words which are denoted as $\{x_i\}_{i=1}^l$, the corresponding word embeddings $\{\mathbf{x}_i\}_{i=1}^l$ are derived using the embedding matrix $\mathbf{E}$. Then, RCNN obtains their corresponding hidden vectors $\{\mathbf{h}_i\}_{i=1}^l$ using the convolution operation and gating mechanism. After obtaining hidden vectors, RCNN uses a pooling operation to get fixed-sized vector presentation, which is fed into softmax layer to finish the prediction. The n-gram convolution operation and gating decay are described as follows:

$$\begin{aligned}
\lambda_t &= \sigma(\mathbf{W}^\lambda \mathbf{x}_t + \mathbf{U}^\lambda \mathbf{h}_{t-1} + \mathbf{b}^\lambda),\\
\mathbf{c}_t^{(1)} &= \lambda_t \odot \mathbf{c}_{t-1}^{(1)} + (1-\lambda_t) \odot (\mathbf{W}_1 \mathbf{x}_t),\\
\mathbf{c}_t^{(2)} &= \lambda_t \odot \mathbf{c}_{t-1}^{(2)} + (1-\lambda_t) \odot (\mathbf{c}_{t-1}^{(1)} + \mathbf{W}_2 \mathbf{x}_t),\\
&\cdots,\\
\mathbf{c}_t^{(n)} &= \lambda_t \odot \mathbf{c}_{t-1}^{(n)} + (1-\lambda_t) \odot (\mathbf{c}_{t-1}^{(n-1)} + \mathbf{W}_n \mathbf{x}_t),\\
\mathbf{h}_t &= \tanh(\mathbf{c}_t^{(n)} + \mathbf{b}),
\end{aligned}$$

where $\mathbf{W}^\lambda$, $\mathbf{U}^\lambda$, $\mathbf{b}^\lambda$, $\mathbf{b}$ and $\mathbf{W}_*$ are learnable parameters, σ is sigmoid function which rescales the value into (0, 1), $\odot$ is dot product, λ_t is gating value determining how much information of $\mathbf{x}_t$ and previous patterns is added into the hidden vector, $\mathbf{c}_t^{(i)}$ refer to the vector for accumulated previous patterns which are ended with x_t include i consecutive tokens. When $\lambda_t = 0$, the convolution becomes a standard n-gram convolution.

We also can build a deep RCNN by adding several convolution layer on top of hidden vectors derived from the bottom convolution layer. Here we consider the RCNN with d convolution layers, which outputs $\{\mathbf{h}_i^d\}_{i=1}^l$. Then, a last pooling operation is conducted on hidden vectors to obtain text representation $\mathbf{r}$. Finally, text representation is fed into a softmax layer. The softmax layer outputs the probability distribution over $|\mathcal{Y}|$ categories for the distributed representation, which is defined as:

$$\mathbf{p}(\mathbf{r}) = \text{softmax}(\mathbf{W}_k^{class} \mathbf{r}).$$

The cross-entropy objective function is used to optimize the RCNN model.

2.2 Prediction

We learn different RCNN models with different embedding sets as input. Formally, we have s embedding sets which are denoted as $\{\mathbf{E}^1, \mathbf{E}^2, \cdots, \mathbf{E}^s\}$, and feed them into s RCNN models, then learn RCNN models separately. We predict sentiment label of testing tweet based on these learned RCNN models, which are described by following functions:

Sets	Corpus	Scale	Algorithm	Dimension	Source	Vocab
gloveG	General	840B	GloVec	300	**R**	2.2M
gloveT	Twitter	27B	GloVec	200	**R**	1.2M
word2vecGN	Google News	100B	Word2Vec	300	**R**	3.0M
word2vecY	Yelp Reviews	0.3B	Word2Vec	300	**S**	0.2M
Ensemble	-	-	-	-	-	5.4M

Table 1: Statistics of the embedding sets. **R** means the embedding set is publicly released and **S** means the embedding set is self-contained. GloVec (Mikolov et al., 2013) and Word2Vec (Pennington et al., 2014) are most popular embedding algorithms. Scale means the size of tokens in corpus, M and B refer to million and billion respectively. The embedding set word2vecY are trained by Word2Vec with default settings and Yelp reviews are available at https://www.yelp.com/dataset_challenge.

Dataset	#num	#category_ratio
Previous SemEvals	50032	1.5/4.7/3.8
SemEval 2017 Test	12284	3.2/4.8/2.0

Table 2: Statistics of datasets.

$$\begin{aligned}
\mathbf{p}_1 &= \mathrm{RCNN}_1(\{x_i\}_{i=1}^{l}, \mathbf{E}^1),\\
\mathbf{p}_2 &= \mathrm{RCNN}_2(\{x_i\}_{i=1}^{l}, \mathbf{E}^2),\\
&\ldots,\\
\mathbf{p}_s &= \mathrm{RCNN}_s(\{x_i\}_{i=1}^{l}, \mathbf{E}^s),\\
\mathbf{p}^{'} &= \sum_{1\leq i\leq s} \mathbf{p}_i.\\
y &= \mathrm{argmax}_{1\leq i\leq |\mathcal{Y}|}\mathbf{p}_i^{'},
\end{aligned}$$

where y is the predicted label.

3 Experiment

3.1 Datasets and Settings

We use 4 embedding sets which are described in Table 1. Meanwhile, we crawl and merge all annotated datasets of previous SemEvals, and split them into training, development, and testing sets with ratio 8:1:1, which are shown in Table 2 together with testing set of SemEval 2017. From the table, we can see that testing set of SemEval 2017 has big differences on the category ratio (negative:neutral:positive), compared with the previous SemEval datasets.

For the model settings, all RCNN models have same configurations but different word embedding sets. We set dimensions of hidden vectors to 250 and depths d to 2. To avoid model over-fitting, we use dropout and regularization as follows: (1) the regularization parameter is set to *1e-5*; (2) the dropout rate is set to 0.3, which is applied in the final text representation. All parameters are learned by Adam optimizer (Kingma and Ba, 2014) with the learning rate 0.001. Note that, all word embedding sets are fixed when training. All models are tuned by the development set in Training.

3.2 Results and Analysis

In this section, we first report the results on datasets of previous SemEvals, which are described in Table 3. Then, we report the performances of our method on SemEval 2017 in Table 4.

From the Table 3, we observe that gloveT performs worst though it is trained on in-domain twitter dataset and the word2vecY performs best though it is derived from yelp reviews. As far as we known, Yelp data is constructed by carefully filtering and is high-quality. Thus, we can include that the quality of corpus is also important as the size of corpus and domain in twitter sentiment classification. Additionally, we can infer that word2vecGN outperforms others in recall of negative category, word2vecY performs best in recall of neutral category, and gloveT is best in recall of positive category. Different embedding sets propose different characteristics. Additionally, the ensemble method obtains a significant improvement of 4%.

In the Table 4, we compare our method with best and median systems in SemEval 2017, and report the results of individual embedding sets. Our method outperforms other systems in accuracy, but performs worse in R_Average, especially in R_Negative. Compared with the median system, our method has improvements of about 5% in both accuracy and R_Average. Different from the results in Table 3, the word2vecY performs

Embeddings	Accuracy	R_Negative	R_Neutral	R_Positive	R_Average
gloveG	70.6	66.3	66.4	77.7	70.1
gloveT	68.3	66.8	61.2	77.9	68.7
word2vecGN	70.5	70.2	68.3	73.5	70.7
word2vecY	72.2	65.0	71.3	76.2	70.8
Ensemble	**74.6**	**72.1**	**71.2**	**80.0**	**74.5**

Table 3: Results on datasets of previous SemEval. R_* means recall value.

Embeddings	Accuracy	R_Negative	R_Neutral	R_Positive	R_Average
Best_system	65.1	**82.9**	51.2	**70.2**	**68.1**
Median_system	61.6	53.1	**65.0**	67.4	61.8
gloveG	62.9	63.0	61.3	66.7	63.7
gloveT	63.7	70.5	57.4	68.2	65.4
word2vecGN	62.9	68.4	60.0	60.8	63.1
word2vecY	61.6	59.1	63.8	60.5	61.1
Our	**66.4**	69.8	64.0	66.8	66.9

Table 4: Results on SemEval 2017. The median system is the system of rank 19th among 38 teams.

worse among these embedding sets, while the gloveT obtains best performances. Additionally, we can observe that gloveT performs best both in R_Negative and R_Positive, and word2vecY performs best in R_Neutral. Compared with the embedding baselines, our ensemble method obtains improvements of 2.7% and 1.5% in accuracy and R_Average respectively, which demonstrates the effectiveness of the proposed method.

3.3 Error Analysis

In this section, we analyze the incorrect predictions of our system in SemEval 2017.

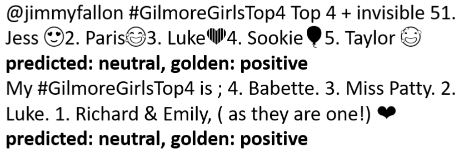
We summarize four kinds of errors in our system. The first one is that some decisions need domain knowledge, which our method only can learn from the labeled datasets. The instances are as follows:

Messi's 100 international goals for Barcelona #fcblive https://t.co/fMkglvusL1 [via @thereisagenius]. Predicted label: **neutral**, golden label: **positive**

#Trudeau gives your cash to #Terrorist #Hamas-influenced group - #UNRWA - @CandiceMalcolm https://t.co/5i5o2qwRWl Predicted label: **neutral**, golden label: **negative**

Messis 9 goals in CL are more than 20 of the 32 teams in the competition have scored in total, and hes tied with five other sides #fcblive Predicted label: **neutral**, golden label: **positive**

The second one is emoticons in tweet, as most of word embedding sets do not include emoticon embeddings and emoticons are always with sentiments. The instances are described in Figure 2



@jimmyfallon #GilmoreGirlsTop4 Top 4 + invisible 51. Jess 😍2. Paris😂3. Luke💜4. Sookie🎈5. Taylor 😂
predicted: neutral, golden: positive
My #GilmoreGirlsTop4 is ; 4. Babette. 3. Miss Patty. 2. Luke. 1. Richard & Emily, (as they are one!) ❤
predicted: neutral, golden: positive

Figure 2: Emoticon instances.

The third one is that sentiments are not consistent in sentences. For example, the first half part is positive, while the second half part is negative, in this case, our system would predict 'positive' or 'negative', the golden label is neutral.

@jimmyfallon 1. Emily 2. Michel 3. Kirk 4. TJ. Love the quirky ones and Emily coz she's such a BIATCH! #gilmoregirlstop4. predicted label **positive**, golden label: **neutral**

The fourth one is the sarcasm, such as: *#Hamas leader: #Trump may be a #Jew https://t.co/jGFZTvj2pF.* predicted label **positive**, golden label: **negative**

4 Conclusion

We propose a simple and effective ensemble method to boost the neural twitter sentiment classification. By using different embedding sets, the system can cover more words and encode more sentiment information. The results on datasets of previous SemEval and SemEval 2017 show the effectiveness of our method. Moreover, error analysis is conducted to propose the main challenges for our method. We release our code for system duplicability.

References

Jan Deriu, Maurice Gonzenbach, Fatih Uzdilli, Aurélien Lucchi, Valeria De Luca, and Martin Jaggi. 2016. Swisscheese at semeval-2016 task 4: Sentiment classification using an ensemble of convolutional neural networks with distant supervision. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*. pages 1124–1128. http://aclweb.org/anthology/S/S16/S16-1173.pdf.

Li Dong, Furu Wei, Yichun Yin, Ming Zhou, and Ke Xu. 2015. Splusplus: A feature-rich two-stage classifier for sentiment analysis of tweets. *SemEval-2015* page 515.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR* abs/1412.6980. http://arxiv.org/abs/1412.6980.

Tao Lei, Hrishikesh Joshi, Regina Barzilay, Tommi S. Jaakkola, Kateryna Tymoshenko, Alessandro Moschitti, and Lluís Màrquez. 2016. Semi-supervised question retrieval with gated convolutions. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*. pages 1279–1289. http://aclweb.org/anthology/N/N16/N16-1153.pdf.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*. pages 3111–3119. http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the 7th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2013, Atlanta, Georgia, USA, June 14-15, 2013*. pages 321–327. http://aclweb.org/anthology/S/S13/S13-2053.pdf.

Preslav Nakov, Alan Ritter, Sara Rosenthal, Fabrizio Sebastiani, and Veselin Stoyanov. 2016. Semeval-2016 task 4: Sentiment analysis in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*. pages 1–18. http://aclweb.org/anthology/S/S16/S16-1001.pdf.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*. pages 1532–1543. http://aclweb.org/anthology/D/D14/D14-1162.pdf.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, Vancouver, Canada, SemEval '17.

Aliaksei Severyn and Alessandro Moschitti. 2015. UNITN: training deep convolutional neural network for twitter sentiment classification. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015*. pages 464–469. http://aclweb.org/anthology/S/S15/S15-2079.pdf.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Baltimore, Maryland, pages 1555–1565. http://www.aclweb.org/anthology/P14-1146.

Yichun Yin, Furu Wei, Li Dong, Kaimeng Xu, Ming Zhang, and Ming Zhou. 2016. Unsupervised word and dependency path embeddings for aspect term extraction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. pages 2979–2985. http://www.ijcai.org/Abstract/16/423.