ECNU at SemEval-2016 Task 7: An Enhanced Supervised Learning Method for Lexicon Sentiment Intensity Ranking

Feixiang Wang[1], **Zhihua Zhang**[1], **Man Lan**[1,2*]
[1]Department of Computer Science and Technology,
East China Normal University, Shanghai, P.R.China
[2]Shanghai Key Laboratory of Multidimensional Information Processing
51151201049@ecnu.cn, 51131201039@ecnu.cn, mlan@cs.ecnu.edu.cn*

Abstract

This paper describes our system submissions to task 7 in SemEval 2016, i.e., Determining Sentiment Intensity. We participated the first two subtasks in English, which are to predict the sentiment intensity of a word or a phrase in English Twitter and General English domains. To address this task, we present a supervised learning-to-rank system to predict the relevant scores, i.e., the strength associated with positive sentiment, for English words or phrases. Multiple linguistic and sentiment features are adopted, e.g., *Sentiment Lexicons*, *Sentiment Word Vectors*, *Word Vectors*, *Linguistic Features*, etc. Officially released results showed that our systems rank the 1st among all submissions in English, which proves the effectiveness of the proposed method.

1 Introduction

The study of sentiment analysis is increasingly drawing attention of Natural Language Processing (NLP). Many of the top performing sentiment analysis systems rely on sentiment lexicon (Tan et al., 2008; Na et al., 2009; Mohammad et al., 2013). A sentiment lexicon is a list of words and phrases, such as *"excellent"*, *"awful"* and *"not bad"*, each of them is assigned with a positive or negative score reflecting its sentiment polarity and strength (Tang et al., 2014a). Higher scores indicate stronger sentiment strength. However, many existing manually generated sentiment lexicons consist of lexicons with only sentiment orientation rather than sentiment strength. For example, the words in *BL* (Ding et al., 2008) are generally divided to two classes, i.e., positive and negative. Although several sentiment lexicons have assigned discrete labels for terms, i.e., *strong* and *weak*, for example, MPQA (Wiebe et al., 2005), there is no continuous real-valued scores to indicate the intensity of sentiment so far.

The task of Determining Sentiment Intensity of English and Arabic Phrases intends to automatically create a sentiment lexicon with real-valued scores indicating the intensity of sentiment. The purpose of this task is to test the ability of an automatic system to predict a sentiment intensity score for a word or a phrase. Phrases include negators, modals, intensifiers, and diminishers. Given a list of terms, the participants are required to assign appropriate scores between 0 and 1, to indicate their strength of association with positive sentiment. The task contains three subtasks (the first two are in English and the third is in Arabic) and we participated the first two subtasks in English. The first General English Sentiment Modifiers Set contains phrases formed by a word and a modifier, where a modifier can be a negator, an auxiliary verb, a degree adverb, or even a combination of those above modifiers, e.g., *"would be very easy"*, *"did not harm"*, and *"would have been nice"*. The second English Twitter Mixed Polarity Set contains phrases made up of opposite polarity terms, such as *"lazy sundays"*, *"best winter break"*, *"couldn't stop smiling"*, etc. The official evaluation measure is Kendall correlation coefficient (Lindskog et al., 2003).

In previous work, the task was treated as a regression problem, the word embedding is used as a feature (Amir et al., 2015). In addition, (Hamdan et al., 2015) adopted unsupervised approach by using

Proceedings of SemEval-2016, pages 503–508,
San Diego, California, June 16-17, 2016.

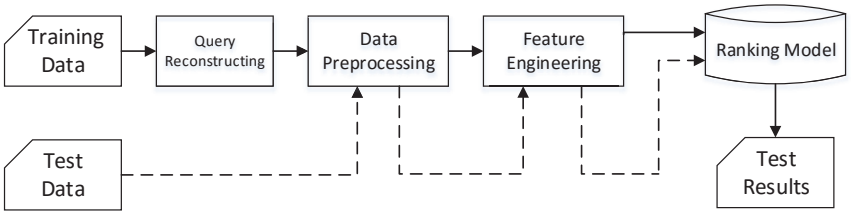

Figure 1: The framework of our proposed system.

several sentiment lexicons for computing the score for each twitter term and ranked them. In this paper, we treated this task as a ranking problem, and used pair-wise strategy to train the model.

The rest of the paper is organized as follows. Section 2 elaborates the procedure of query reconstruction, data preprocessing, feature engineering and the learning-to-rank model built in our systems. Section 3 describes the data sets and experiments. Finally, Section 4 concludes this work.

2 System Description

The purpose of this task is to predict the sentiment strength of given words or phrases, it is reasonable to regard the value of strength as the relevant score refer to positive polarity. Thus, to address the sentiment strength prediction task, we presented a supervised learning-to-rank system to predict the relevant score which interprets the strength associated with positive sentiment. Figure 1 depicts the architecture of our system, which contains four main modules, i.e., *Query Reconstruction*, *Data Preprocessing*, *Feature Engineering*, and *Ranking Model*. The diversity of methods for General English and English Twitter domains is located in the different training data.

2.1 Query Reconstruction

Since the sentiment strength task was treated as a ranking problem, we converted the provided training data into appropriate forms as the input of our ranking system in the first stage. In general, several queries were fed into the ranking system served as training data. In consideration of the provided training data is a word list ordered by the strength score associated with positive sentiment, we manually divided the word list into several *"queries"* and the words in each query were sorted by their corresponding scores. Specifically, the test data would not be processed into *Query Reconstruction* module, because we considered all records in the test set as a unique query and the *Ranking Model* was used to predict their relevant scores.

2.2 Data Preprocessing

Due to the characteristic of training data in English Twitter domain, which consists of many informal words, such as *"waiiiiit"*, *"#happytweet"*, etc., we converted the irregular forms to normal forms. For example, the elongated word *"waiiiiit"* was transformed into *"wait"*, the hashtag *"#happytweet"* was converted into *"happy tweet"*. Then, we used the processed data to perform lemmatization and stemming to extract the further information with the aid of *NLTK tool*[1].

2.3 Feature Engineering

To address the sentiment strength task, we extracted features summarized as follows: *Sentiment Lexicon Features*, *Sentiment Word Vectors*, *Word Vectors*, and *Linguistic Features*.

2.3.1 Sentiment Lexicon Features

We employed the following seven sentiment lexicons to extract *Sentiment Lexicon Features*: *Bing Liu lexicon*[2], *General Inquirer lexicon*[3], *IMDB*[4], *MPQA*[5], *AFINN*[6], *NRC Hashtag Sentiment Lexicon*[7], and *NRC Sentiment140 Lexicon*[8]. Generally, we

[1] http://www.nltk.org/

[2] http://www.cs.uic.edu/liub/FBS/sentiment-analysis.html#lexicon

[3] http://www.wjh.harvard.edu/inquirer/homecat.htm

[4] http://anthology.aclweb.org//S/S13/S13-2.pdf#page=444

[5] http://mpqa.cs.pitt.edu/

[6] http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010

[7] http://www.umiacs.umd.edu/saif/WebDocs/NRC-Hashtag-Sentiment-Lexicon-v0.1.zip

[8] http://help.sentiment140.com/for-students/

transformed the sentiment orientation of all words in all sentiment lexicons into the range of -1 to 1, where the minus sign denotes negative sentiment and the positive number indicates positive sentiment.

We considered each term (consisting of phrases and words) in the provided data as a tuple. If one term contains a single word, the size of this tuple is 1. If one term is a phrase, e.g., *"happy accident"*, the tuple size is the count of words in the phrase (for *"happy accident"*, it is 2). Then, for each tuple, we concatenated the following sentimental scores extracted from each sentiment lexicon: (1) the maximum sentiment score of words, (2) the minimum sentiment score of words, (3) the sum of sentiment scores of all words. If one word does not exist in a sentiment lexicon, its corresponding score is set to 0. Specifically, if any negator, e.g.,"*not*", exists in a tuple, we reverse its sentiment orientation.

Considering the diversity of word forms, we also extracted additional sentiment lexicon features about their lemmatization and stemming forms. The final sentiment lexicon representation of the i-th record $l_i \in \mathbb{R}^{n_l * n_s * t}$, where n_l is the number of sentiment lexicons (i.e.,7), n_s is set as 3 representing the three scores calculated according to the above rules, t is also set as 3 representing the three forms (i.e., original, lemmatization and stemming forms) of the word in the record.

2.3.2 Word Vectors

Word vector is a continuous-valued representation of the word which carries syntactic and semantic information. In this task, we adopted various *Word Vectors* as features. Since the task data contains phrases made up of more than one word, it is necessary to convert these word vectors into a phrase vector. Unlike previous work which summed up all words vectors to represent the phrase, we first check whether the phrase is present in training data or not. If a phrase is present in training data, we actually treat it as a single token and learn its vector based on the given word vector learning model. If it is a new phrase not existing in training data, we follow previous work and sum up all vectors of words in this phrase as a phrase vector.

In this part, we introduce two types of word vector which are adopted in our method, i.e., Traditional Word Vector and Sentiment Word Vector, which are described as follows:

Traditional Word Vectors:

- *GoogleW2V*: We used the publicly available *word2vec* tool[9] to get word vectors with dimensionality of 300, which were trained on 100 billion words from Google News as *GoogleW2V* feature.

- *GloveW2V*: The Glove(Pennington et al., 2014) rely on different assumptions about the relations between words within a context window. We used the available pre-trained word vector with dimensionality of 100 and trained on 2 billion tweets, which were supplied in *GloVe*[10] as *GloveW2V* feature.

- *NormalW2V*: We fed *NRC140 tweet corpus* into *word2vec tool* to build *NormalW2V* with dimensionality of 100.

- *NormalW2V_Multi*: To train specific word vectors (i.e., words and phrases) for this task, we preprocessed the *NRC140 corpus* by connecting words in the phrase appeared in training data to abtain a single token. The processed data were fed into word vector model to train *NormalW2V_Multi* with dimensionality of 100.

Sentiment Word Vectors: Since the above word vectors are trained based on the context, they are supposed to contain semantic and syntactic information. However, due to lack of sentiment information, these traditional word vectors may not be quite effective for sentiment analysis tasks. To address this issue, our previous work and other researchers has proposed methods to learn sentiment word vectors.

- *SWV*: In (Zhang and Lan, 2015) we proposed the *Combined-Sentiment Word Embedding Model* (i.e., *SWV-C*) to learn sentiment word vectors, which are confirmed to be helpful to settle sentiment analysis task. In this work, we used this model to train the *SWV* with the aid of *NRC140 tweet corpus* (Go et al., 2009), where the corpus is made up of 1.6 million positive tweets and 1.6 million negative tweets. The vector size is set as 100.

[9] https://code.google.com/archive/p/word2vec

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

- *SWV_Multi*: Similar with *NormalW2V_Multi*, we used the *NRC140 corpus* as training corpus and reconstructed it to the task specific form. The processed corpus was employed as input of *SWV-C* model to generate *SWV_Multi* with dimensionality of 100.

- *SSWE*: The sentiment-specific word embeddings (*SSWE*) were proposed by (Tang et al., 2014b), which is quite similar to our idea in (Zhang and Lan, 2015) but differs in proposed models. This *SSWE* sentiment word embeddings were trained by using multi-hidden-layers neural network with vector size of 50.

2.3.3 Linguistic Features

- Negation: The sentiment polarity of word or phrase can be reversed by a modification of negation. Therefore, we collected 29 negations from Internet and we sign 1 or 0 to this binary feature if corresponding negation is present or absent in the pending word or phrase.

- Elongated: This feature represents if word or phrase with one character repeated more than twice, e.g.,*"lottttt"*.

2.4 Ranking Model

Different from the classification or regression methods, which focus on labeling the single record, the ranking method takes the relation between two arbitrary records associated with a query into consideration.

In this module, we used a supervised learning-to-rank approach to perform ranking. Generally, the mentioned approach can be divided into three groups: point-wise, pair-wise, and list-wise. We adopted the second strategy, i.e., pair-wise for our work. In pair-wise strategy, several record1-record2 pairs were constructed with a query and some records which were provided in advance. If record1 is more relevant than record2 in terms of the given query, this pair label will be set as 1, otherwise 0.

3 Experiments and Results

3.1 Datasets

This sentiment strength prediction task was severed as the subtask E of Sentiment Analysis on Tweet in SemEval 2015. Thus, the trial (i.e., 15*trial*) and test (i.e., 15*test*) data in SemEval 2015, which contained 200 and 1,315 records separately, are integrated as training data for this task. The organizer provided 200 records as development data set for each domain (i.e., 16*trial_Twitter* and 16*trial_General*) and the labels are the same as before, where each record is labeled with a decimal in the range of 0 to 1 and the score is the strength associated with positive sentiment.

In consideration of the lack of training data, we expanded it with the Language Assessment by Mechanical Turk lexicon (i.e., *LabMT*) which automatically labeled by (Dodds et al., 2011). It contains 10,222 words rated on a scale of 1(sad) to 9(happy). Note that the labels in the *LabMT* are different from the standard data, we converted the score to the scale of 0 to 1 by min-max normalization, i.e., $\frac{x-min}{max-min}$.

3.2 Evaluation Metrics

For this task, Kendall rank correlation coefficient (usually measures the association between two measured quantities) is used as the metric to compare the ranked lists. Besides, the scores for Spearman's Rank Correlation(a nonparametric measure of statistical dependence between two variables) is provided as well. The Kendall rank correlation coefficient is severed as the final official evaluation criteria.

3.3 Experiment on training data

As we described in 2.1, several operations should be conducted to transform the raw data form to accommodate the ranking system. Considering that the *LabMT* is a term list that has been automatically labeled, while the provided standard data is more precise, so we constructed each query in *LabMT* with 200 records. With regard to the provided standard data (i.e., 15*trial*, 15*test*), each query was made up of 20 records.

To construct training data for English Twitter domain, the *LabMT*, 15*train* and 15*test* were utilized, whereas the 16*trial_Twitter* was utilized as development data. Several types of features have been proposed in 2.3, in order to conduct feature selection, we adopted *hill climbing* which is described as: keeping adding one type of feature at a time until no further improvement can be achieved.

The system's diversity of two domains lies on

Feature		English Twitter	General English
Traditional W2V	GoogleW2V	√	
	NormalW2V	√	√
	NormalW2V_Multi		
	GloveW2V		√
Sentimental W2V	SWV	√	√
	SWV_Multi		√
	SSWE	√	√
Sentiment Lexicon	SentiLexi	√	√
Linguistic	Negation	√	√
	Emphasize	√	−
Ranking Results (Kendell/Sperman)		**59.83%/75.38%**	**71.67%/86.83%**
Regression Results (Kendall/Sperman)		57.80%/73.01%	70.50%/85.26%

Table 1: Results of feature selection experiments for Twitter English and General English domains.

the different training data we utilized: all tweet-related data were adopted in English Twitter, while the words and phrases with *hashtag*(#) or informal forms were removed for General English. Thus, the filtered data of *LabMT*, *15trial* and *15test* were severed as training data and the *16trial_General* were utilized as development data. The process of feature selection was similar with English Twitter domain except that the *Emphasize* feature was not used.

Table 1 shows the results of feature selection experiments on the development data, which lists the optimal feature sets of two domains.

From Table 1, it is interesting to find: (1) The *Sentiment Lexicon* features make a considerable contribution, because that the used sentiment lexicons contain sentiment information to some extent and its sentiment scores in lexicons are closely related to the strength associated with positive sentiment. For example, in *BL*, the scores of positive words are 1 and the negative words are represented as -1. (2) The *Linguistic* features (i.e., *Negation* and *Emphasize*) also contribute to performance. As for *Negation*, the possible reason may be that there are plentiful negators existed in training and development data and the emphatic words have similar situation. (3) The *NormalW2V* and *SWV* are both effective features for this task. In our further experiments which test the *SWV* and *NormalW2V* respectively, we found that *SWV* performs much better than *NormalW2V*, which showed that the *Combined Sentiment Word Vector Model* indeed captured sentiment information from abundant auto-labeled tweets. (4) The ranking based system outperforms the regression method, which indicates that it is convincing to regard this labeling task as a ranking problem. (5) Compared with the results on English Twitter domain, we notice that the performance is much better. Based on the observation on development data of two domains, we found that the phrases on English Twitter domain are made up of opposite polarity terms, e.g., *"happy accident"*, *"couldn't stop smiling"*, while the records in General English domain are much more ordinary. The diversity of data distribution results in the above mentioned gap.

3.4 System Configuration

We built two systems for the two subtasks. In our preliminary experiments, we examined several algorithms with different parameters implemented in *RankLib tool*[11], e.g., *Random Forest*, *RankNet*, *RankBoost*, and *ListNet*. According to the best performance in our experiments, we adopted *Random Forest* algorithm with parameters $tree = 5$, $bag = 100$ for English Twitter domain and $tree = 1$, $bag = 300$ for General English domain for our final submitted systems.

3.5 Results and Analysis

Table 2 lists the performances of our system and the top ranked system provided by organizer on test data.

The results in the Table 2 show that our system indeed performed well on sentiment strength prediction task regardless of domains, where the proposed system ranks 1st above both domains among all submissions. The results of test data are consistent with our experiment on training data, the performance on

[11] https://people.cs.umass.edu/ vdang/ranklib.html

English Twitter domain is worse than that on General English domain due to the data distribution.

Domain	Rank	TeamID	Kendall(%)	Sperman(%)
General	1	ECNU	**70.42**	**86.27**
	2	UWB	65.91	85.36
	3	LSIS	34.97	50.75
Twitter	1	ECNU	**52.31**	**67.40**
	2	LSIS	42.16	59.06
	3	UWB	41.38	57.82

Table 2: Performances of our systems and the top-ranked system for two domains. *General* and *Twitter* stand for *General English domain* and *English Twitter domain* respectively.

4 Conclusion

In this paper, we presented a supervised learning-to-rank system to predict the strength of positive sentiment associated with words and phrases. Multiple features, e.g., *Sentiment Lexicon*, *Sentiment Word Vectors*, *Word Vectors*, and *Linguistic Features*, were presented. We find that the *Sentiment Lexicon Features* and the *Sentiment Word Vectors* make contributions to performance improvement. However the phrase vector is not effective as expected and in future work it is interesting to explore more ways to represent phrase.

Acknowledgements

This research is supported by grants from Science and Technology Commission of Shanghai Municipality (14DZ2260800 and 15ZR1410700), Shanghai Collaborative Innovation Center of Trustworthy Software for Internet of Things (ZF1213).

References

Silvio Amir, Ramón Astudillo, Wang Ling, Bruno Martins, Mario J. Silva, and Isabel Trancoso. 2015. Inesc-id: A regression model for large scale twitter sentiment lexicon induction. In *SemEval 2015*, pages 613–618, Denver, Colorado, June. Association for Computational Linguistics.

Xiaowen Ding, Bing Liu, and Philip S Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 231–240. ACM.

Peter Sheridan Dodds, Kameron Decker Harris, Isabel M Kloumann, Catherine A Bliss, and Christopher M Danforth. 2011. Temporal patterns of happiness and information in a global social network: Hedonometrics and twitter. *PloS one*, 6(12):e26752.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, pages 1–12.

Hussam Hamdan, Patrice Bellot, and Frederic Bechet. 2015. Lsislif: Feature extraction and label weighting for sentiment analysis in twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 568–573, Denver, Colorado, June. Association for Computational Linguistics.

Filip Lindskog, Alexander Mcneil, and Uwe Schmock. 2003. *Kendall' s tau for elliptical distributions*. Springer.

Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242*.

Seung-Hoon Na, Yeha Lee, Sang-Hyob Nam, and Jong-Hyeok Lee. 2009. Improving opinion retrieval based on query-specific sentiment lexicon. In *ECIR*, volume 9, pages 734–738. Springer.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Songbo Tan, Yuefen Wang, and Xueqi Cheng. 2008. Combining learn-based and lexicon-based techniques for sentiment detection without using labeled examples. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 743–744. ACM.

Duyu Tang, Furu Wei, Bing Qin, Ming Zhou, and Ting Liu. 2014a. Building large-scale twitter-specific sentiment lexicon: A representation learning approach. In *Proceedings of COLING*, pages 172–182.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014b. Learning sentiment-specific word embedding for twitter sentiment classification. In *The Annual Meeting of the Association for Computational Linguistics*, pages 1555–1565.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210.

Zhihua Zhang and Man Lan. 2015. Learning sentiment-inherent word embedding for word-level and sentence-level sentiment analysis. In *2015 International Conference on Asian Language Processing, IALP*.