EasyChair Preprint

№ 538

Sentiment Classification of Chinese Financial Reviews

Fahai Zhong, Hang Yuan, Mengnan Li and Ronghui Luo

September 28, 2018

Sentiment Classification of Chinese Financial Reviews

Fahai Zhong
School of Physics and Engineering
Zhengzhou University
Zhengzhou, China 450001
Email: fahai.zhong@stu.zzu.edu.cn

Hang Yuan
School of Physics and Engineering
Zhengzhou University
Zhengzhou, China 450001
Email: hang8756@foxmail.com

Mengnan Li
School of Business
Zhengzhou University
Zhengzhou, China 450001
Email: 517969770@qq.com

Ronghui Luo*
School of Physics and Engineering
Zhengzhou University
Zhengzhou, China 450001
Email: luoji666@zzu.edu.cn

***Abstract*—In order to explore the method of sentiment classification of Chinese text, this paper takes the Chinese comments of financial sector as the main research object and carries out text sentiment classification tasks. This paper proposes to initialize the word embedding with dictionary, to solve the problem that some words have opposite sentimental trendencies but similar distributed representation. Three commonly used evaluation indexs and their average value(AvgScore), the accuracy of classifier, the average recall rate and MacroF1, were given to evaluate the model. Through comparative analysis to explore the factors that influence the effect of classifier, this paper uses CNN, LSTM, GRU as the substructer and constructs a total of 9 models with different structures and depths for comparative study. Based on the top three models on AvgScore, this paper studies the method of initializing the word embedding with sampling and random perturbation technique. The results show that the sampling technique has the greatest impact on the classifier effect. Among the different sampling techniques, Avg Score difference by 1.1% to 38.3%. The best results can be obtained from the mix use of downsampling technique in majority and few oversampling technique. The classifier obtained by word embedding with sentiment dictionary is better than using other word vectors. Its highest accuracy rate, MacroF1, Avg Score respectively are 82.37%, 77.26% and 77.62%, and the other highest are 82.19%, 76.73%, 77.08%. In this paper, the top three classifiers with the highest Avg Score are selected to build the classifier in ensemble approach. The accuracy of the final ensemble method classifier is 84.00%, the average recall is 74.58%, the Macro F1 is 79.50%, and the Avg Score is 79.36%.**

***Index Terms*—text classification, sampling method, word embedding initialization method,Ensemble methods, imbalanced sample classification.**

I. Introduction

Text sentiment classification, also known as text sentiment analysis, is an important subfield in natural language processing. It is the analysis, processing, induction and reasoning of subjective text with sentimental color.As network is more and more popular, people always like to publish their comments on some products on the web. These comments tend to have strong sentimental tendency, which is important for companies to improve their products and services. sentimental analysis techniques are also used in government work such as public opinion tracking and security supervision.

Text sentiment classification usually involves testing a text whether it expresses a positive, negative or neutral sentiment; However, in earlier studies, the dichotomy task of text sentiment was generally used because it is much more difficult to classify text sentiment into multiple categories than two categories. On the SemEval:2017:task4([1]) of the International Cognitive Semantics Conference, the average recall rate of tweets can be as high as 88.2%[2], but the maximum is 68.1%[2] when the same text is triaged.

In dealing with the Chinese analysis of sentiment,[3] divided Chinese sentiment analysis research into two categories: The monolingual and the bilingual approach. The former directly carries out the task of sentiment analysis on Chinese, while the latter uses machine translation technology to translate Chinese into English, and then uses mature English sentiment analysis model to conduct the task of sentiment classification. For example, [4] used Google translation to translate Chinese into English, and used the attention-based LSTM model, achieving an accuracy of 82.4% in the task of classifying Chinese text. On the other hand, the previous studies can be divided into rule-based and learning-based methods. Rule-based methods can often achieve a high effect [5], but this requires an sentiment word dictionary and semantic rules defined by experts, which requires a lot of priori knowledge, and sentimental dictionaries and semantic rules set for specific areas may not be transferable to other areas [6]. The learning-based method only needs to build a new training set when the task changes and retrain the model. With the advent of the era of Web2.0, the acquisition of data

sets becomes easier, and the tagging of data labels can be completed by ordinary and cheap labor in a short period of time. With the continuous improvement of computing ability, the deep learning model is more and more widely used in the task of text sentiment classification.

In order to explore the method of Chinese text sentiment classification, this paper takes the Chinese commentary of financial marketing activities as the main research object and constructs the deep learning model of quantifying words to complete the task of three-way classifying text sentiment. The organizational structure of this paper is: section **II** introduces the composition of data set, section **III** introduces text preprocessing, section **IV** introduces the main structure of the model, section **V** introduces the comparative experiment and experimental results in this paper, and section VI summarizes the experimental results.

II. Data set

The data set used in this study is provided by the organizing committee (PAC)[1] of the 2017 Parallel Application Challenge hosted by Intel(China). It is the user comment data of the financial, social, BBS and application webpages that baidu serves as the portal.The final sample distribution is shown in **Table I**.

TABLE I
Data set distribution

Dataset	Positive	Neutral	Negative
Training Data	5253(5.94%)	80663(91.27%)	2462(2.79%)
Test Data	985(29.63%)	1988(59.81%)	351(10.56%)

It can be seen from **Table I** that the proportion of "positive" tag samples in this training set is 5.94%, that of "negative" tag samples is 2.79%, and that of "neutral" tag is 91.27%. There is an obvious unbalanced sample distribution problem. The proportion of samples labeled "positive" in the test set was 29.63%, that of samples labeled "neutral" was 59.81%, and that of samples labeled "negative" was 10.56%.

To test the model effect, PAC provides a test set. It's not hard to see that the sample distribution of the test set and the training set is very different by comparing test set and training set. In this study, there is a feature space inconsistency between test set and training set, so it is very necessary to solve the unbalanced distribution of training samples.

III. Preprocessing

This section introduces the text pretreatment process, mainly including the word segmentation, word vectorization and data sampling. After preprocessing stage, the text sequence of data sample is transformed into a digital sequence that can be identified by a machine.

A. Word Segmentation

Since Chinese is composed of Chinese characters, there is no natural separator to divide sentences into words. If the natural language processing task need to use quantified words, the sentence should be divided into multiple words first. The paper uses the jieba participle tool[2], which has a better effect. [7], [8] also chose to use jieba for Chinese word segmentation.

In response to cacography, slang, informal abbreviations, indirect expressions, etc., a user-defined dictionary was added when the jieba word segmentation tool was invoked.

B. Word Vectorization

[7] studied the text classification effect of different encoding methods in Chinese,Japanese and Korean, and the results showed that word-level coding could still achieve better results without good word segmentation. Therefore, this paper studies the task of text sentiment classification in a lexical quantitative way. The initialization of word vector can be obtained by training unlabeled data sets through word2vec[9].

Corpus: The unlabeled data set is mainly composed of Chinese wikipedia corpus (2017.07.20). The corpus has 1.5GB Chinese text, covers topics in various fields, and has the latest Internet vocabulary, such as "Dama", "mystic energy", "spectators" and so on. It is a relatively comprehensive and cutting-edge corpus.

Word Embedding Initialization with Sentiment Word Dictionary: In previous studies, word vectors are often initialized in two ways: use Word2vec for training and random generation. [10] also compared the two initializations, and the results showed that the word-embedding effect through training was better than randomly generated. The word vector obtained through training can well represent the relation between words, and the cosine similarity of grammatically similar words is high. But because of this, two words that describe the same subject but opposite poles of sentiment will result in the other words in the context are similar, which can lead to similar word embeddings, because of the same syntax structure and description body. According to the test, about 0.79 of the ten most similar words for each positive sentiment word on average trained by Word2vec were negative sentimental words(positive and negative words were obtained from the sentiment dictionary), which showed that some words were of opposite sentimental polarity but similar word vectors. In order to solve this problem, this paper proposes a method to construct new word vectors by combining sentiment dictionary: Using the sentiment dictionary of Taiwan university[11] to construct new word vectors through the "90+10" mode. That is, 90 dimensions of the word vector are acquired through training, and the other 10 dimensions are obtained through the sentiment dictionary. If a word

[1]http://www.pac-hpc.com

[2]https://github.com/fxsjy/jieba

appears in positive sentimental dictionary, then its 10 dimensions of sentiment will be taken a random value near 1. If it is in the negative sentimental dictionary, its sentimental degree will take a random value near -1. If not, take a random value near 0.

C. Sampling Method

In the training set, only 5.94% of the samples labeled "positive" and only 2.79% of the samples labeled "negative". Without any sampling performed on the training set, "positive" and "negative" samples will be treated as noise filtering. It is very bad that the classifier will eventually classify all samples as "neutral". So it is necessary to reconstruct the data set.

Many scholars have studied the imbalance of data sets[12], [13], [14], [15]. SMOTE(Synthetic Minority Oversampling Technique)[14] is widely used in the processing of data imbalance samples. SMOTE does not apply here. Storing word vector index is much more efficiently than store the term vectors. To Use SMOTE technology, the preprocessed samples must be transformed from word vector index to word vector index. The time and storage cost will become huge.

Oversampling of the majority and undersampling of the minority[15] is also widely used in deep learning. In order to avoid the possible overfitting problem caused by simply copying samples, a random perturbation method is proposed to make the difference between the duplicated samples.

IV. Model

CNN,LSTM,GRU and other network structures have been widely used in NLP tasks. [16] combined CNN and LSTM, constructed the CNN-LSTM model and carried out the Multi-Dimensional sentiment analysis task, obtained the effect of root mean square error (RMSE) of 0.874. In this paper, these neural network structures commonly used in natural language processing tasks are taken as substructures, and models with different combinations and different depths are constructed. These models are evaluated and analyzed to find the most suitable models for Chinese text sentiment classification.

CNN: Convolutional neural networks (CNN) has been widely used in computer vision [17], [18], natural language processing [10]. According to [18], this paper uses the CNN structure with Dropout and ReLU, which is mainly composed of convolutional layer, pooling layer and fullly connected layer. In the convolution layer, the convolution kernel size is set to 5x1, and in the pooling layer, the kernel size is set to 4x1.

LSTM: Long short-term memory term is a kind of special recurrent neural networks, which was first proposed in 1997[19]. It is one of the most commonly used structures in text sentiment classification . LSTM reads features by forward from f1 to fi, ultimately generates annotations $H = (h_1, h_2, ...h_T)$, where h_i is the hidden state when time-step is i. h_i summarizes all information from f_1 to f_i. Bidirectional Long short term memory (BiLSTM) is a variant of LSTM[20], in BiLSTM,$h_i = \overrightarrow{h}_i || \overleftarrow{h}_i$.

GRU: [21] proposed the Gated Recurrent Unit (GRU),makes each regular unit adaptively to capture the dependencies of different time scales. Like LSTM unit, GRU has a door control unit that regulates the internal information flow of the unit, but no separate storage units.

Attention: Attention-based model has been widely used in the task of text sentiment classification [4], [22]. Not every word in a sentence expresses sentiment equally. The attention mechanism is used to find out the relative contribution of each word and assign the weight of ai to each word. Finally, the weight of all words is calculated. In form:

$$\begin{cases} e_i = tanh(W_h h_i + b_h), \quad e_i \in [-1, 1] \\ a_i = \frac{exp(e_i)}{\sum_{t=1}^{T} exp(e_t)}, \quad \sum_{t=1}^{T} a_i = 1 \\ r = \sum_{t=1}^{T} a_i h_i, \quad r \in R^{2L} \end{cases}$$

V. Experiments

The experiment was carried out on Intel's open source framework BigDL. Intel(R) Core(TM) i7-6900k CPU in parallel with 8 CPUs was used for training and testing. All model trainings have Adam[23] as the optimizer. Maxepoch is set to 6 and batch sizes are 128. The loss function is cross-entropy error of supervised sentimentclassification. 1/4 samples of the training set are extracted as validation set.

Semeval-2017 [1] presented three evaluation indexes, macroaveraged recal(AvgRec), Accuracy, macroaveraged f1-score (Macro-F1), and AvgRec was used as the main evaluation index. The number of "positive" and "negative" samples in this test set is not large enough, and the gap of the smaller number of samples, which can accurate predict "positive" or "negative", will have a greater impact on AvgRec. Therefore, the The average scores(Avg Score) of AvgRec,Accuracy and Macro-F1 are given in this paper, which is expected to provide a stable reference value for the model.

$$\begin{cases} AvgRec = \frac{Recall_{Positive} + Recall_{Neutral} + Recall_{Negative}}{3} \\ F1 - Score = \frac{2 \times Precision \times Recal}{Precision + Recal} \\ MacroF1 = \frac{F1_{Positive} + F1_{Neutral} + F1_{Negative}}{3} \\ AvgScore = \frac{AvgRec + Acc + MacroF1}{3} \end{cases}$$

Precision is the percentage of accurately predicted instances of this class divided by all instances that are predicted to be this class, and Recall is the ratio of the accurately predicted instances of the class divided by the instances of the class in the data set.

A. Model comparison

The effects of different structures, different depths, one-way or two-way circulatory neural networks on the model were compared.**Table II** shows the results of different models. In the single-structure model, the accuracy of

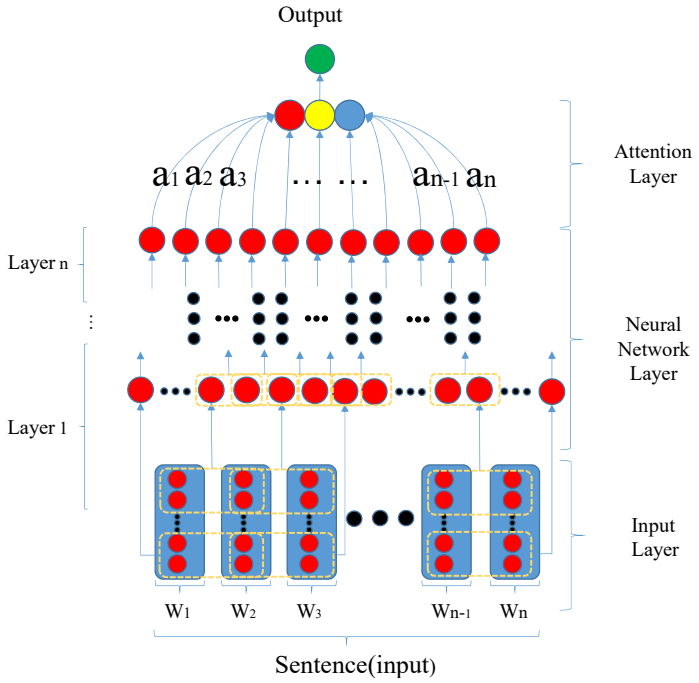


Fig. 1. Model

TABLE II
TEST RESULTS FOR DIFFERENT MODELS.

Model	Accuracy	AvgRec	Macro F1	Avg Score	Time
CNN	81.17%	72.02%	76.35%	76.51%	4min 56s
LSTM	80.39%	71.98%	75.81%	76.06%	57min 33s
GRU	79.15%	69.44%	73.82%	74.14%	42min 40s
CNN-CNN	82.04%	**73.55%**	**77.26%**	**77.62%**	8min 38s
CNN-LSTM	79.72%	70.95%	74.26%	74.98%	49min 12s
CNN-GRU	77.83%	69.05%	72.67%	73.18%	34min 34s
CNN-CNN-CNN	**82.37%**	71.96%	77.01%	77.12%	9min 48s
CNN-LSTM-LSTM	79.06%	69.84%	74.19%	74.36%	1h 27min 40s
CNN-LSTM-GRU	79.87%	70.25%	74.42%	74.85%	1h 24min 12s

TABLE III
DISTRIBUTION OF DIFFERENT SAMPLING METHODS

Sampling method	Positive	Neutral	Negative
No sampling	3931(5.91%)	60753(91.39%)	1796(2.70%)
Undersampling	3932(33.18%)	6070(51.22%)	1848(15.59%)
Mixing sampling	34357(31.69%)	45472(41.94%)	28581(26.36%)
Oversampling	51035(31.05%)	60671(36.92%)	52634(32.03%)

CNN,LSTM and GRU respectively are **81.17%**,80.39% and 79.15%, AvgRec are **72.02%**,71.98% and 69.44%, Macro-F1 are **76.35%**,75.81% and 73.82%, and Avg Score are **76.51%**,76.06% and 74.14% respectively. Compared with other structures, CNN structure is more suitable for the task of sentiment classification in this paper. The model obtained the highest accuracy (82.37%). The CNN-CNN model obtained the highest average recall rate (73.55%), MacroF1(77.26%) and Avg Score(77.62%). The three models with the highest Avg Score are CNN, CNN-CNN and CNN-CNN-CNN.

B. Sampling method

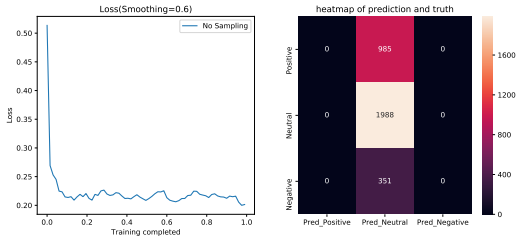
Table III describes the distribution of three Positive, Neutral and Negative kinds of samples, under the condition of no sampling, undersampling, mixsampling and oversampling.

Figure 2 describes the training process and training results when the training set is not sampled. At the beginning of training, Loss dropped rapidly and reached a lower value. However, Loss has been floating around 0.2 since then, and there is no significant downward trend. As can be seen from heatmap, no matter what the input is, the model will be classified as "neutral" in the prediction process, and Positive and Negative have been filtered out as noise.

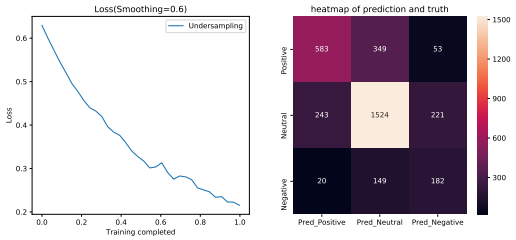
The consequence of unsampling is that Loss falls too fast and leads to local optimum, while undersampling is Loss drops slowly and the model is under-fitting, resulting in low performance of classifier. The Loss of oversampling, and simultaneous oversampling and undersampling is relatively normal. By comparing the heatmap in **Figure 2c** and **Figure 2d**, it can be seen that the simultaneous oversampling and undersampling are the best.

C. Word Embedding Initialization method

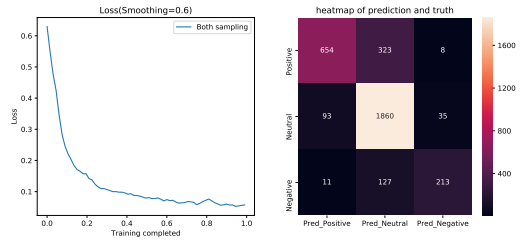
Comparing the models constructed by three word embedding, the random initialization word embedding, corpus word embedding and word embedding combined with the sentiment dictionary, and model structure by CNN, the CNN - CNN, CNN-CNN-CNN, the results are shown in **Table V**. It can be seen that in combination with sen-



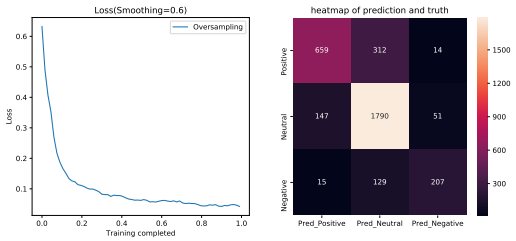
(a) Result of No sampling



(b) Result of Undersampling



(c) Result of Mixing sampling



(d) Result of Oversampling

Fig. 2. The results of different sampling techniques

TABLE IV
RESULTS OF DIFFERENT SAMPLING METHODS

Sampling method	Accuracy	AvgRec	Macro-F1	AvgScore
No sampling	0.5981	0.3333	0.2495	0.3935
Undersampling	0.7674	0.717	0.7122	0.7323
Mixing sampling	0.8204	0.7355	0.7726	0.7762
Oversampling	0.7990	0.7197	0.7476	0.7555

TABLE V
RESULT OF WORD EMBEDDING INITIALIZATION METHOD,THE "SPECIAL" MEANS USING SENTIMENT DICTIONARY TO INITIALIZE.

Model	Accuracy	AvgRec	Macro F1	Avg Score
CNN(Special)	0.8116	0.7202	0.7635	**0.7651**
CNN(Normal)	0.8039	0.7278	0.7523	0.7613
CNN(Rand)	0.7966	0.7207	0.7550	0.7575
CNN-CNN(Special)	0.8204	0.7355	0.7726	**0.7762**
CNN-CNN(Normal)	0.7897	0.7319	0.7352	0.7522
CNN-CNN(Rand)	0.7963	0.7211	0.7521	0.7565
CNN-CNN-CNN(Special)	0.8237	0.7196	0.7701	**0.7712**
CNN-CNN-CNN(Normal)	0.8189	0.7262	0.7673	0.7708
CNN-CNN-CNN(Rand)	0.8219	0.7174	0.7647	0.7680

timent dictionary word embedding obtained the optimal results.

D. Ensemble methods

Ensemble methods[24] are to build a new classifier in the predictive stage by means of a averageing weight vote. Three models with the highest Avg Score were used for Ensemble: CNN with Oversampling, CNN-CNN, and CNN-CNN without using the Random Perturbation Technology. The result is shown in **Table VI**. Ensemble methods in

TABLE VI
RESULT OF ENSEMBLE METHODS,MEANING 'NONE' IS NOT USED RANDOM PERTURBATION TECHNOLOGY.'MIXED' IS A MIXTURE OF OVERSAMPLING AND UNDERSAMPLING

Model	Accuracy	AvgRec	Macro F1	Avg Score
CNN(Oversampling)	0.8222	0.7309	0.7733	0.7755
CNN-CNN(Mixed)	0.8204	0.7355	0.7726	0.7762
CNN-CNN(None)	0.8273	0.7314	0.7757	0.7781
Ensemble	**0.8400**	**0.7458**	**0.7950**	**0.7936**

Accuracy, AvgRec, Macro F1, Avg Scores are 1% to 2% higher than a single model.

VI. CONCLUSION

This paper takes the Chinese commentary of financial marketing activities as the main research object to explore the method of Chinese text sentiment classification task. The Avg Score of Accuracy, AvgRec and Macro F1 were used as the main evaluation indicators to evaluate models of different structures. Among the 9 models with different depth composed of CNN,LSTM and GRU, the CNN-CNN model has the best effect. In RNN structure, using bidirectional structure is not good. In data set reconstruction, the best results can be obtained by using both majority oversampling and minority undersampling techniques. When oversampling and copying samples, generating random perturbation can give better results to some models. The effect of random perturbation technology is not obvious, and the range of random perturbation may need to be further determined through research. When the word embedding is initialized, the word embedding combined with sentiment dictionary proposed in this paper can achieve a better effect. Taiwan university has released the

dictionary they researched[11]. In the past ten years since its emergence, there have been many online buzzwords expressing sentiment, such as facial expressions, etc. In the future work, the sentiment dictionary can be updated to play a greater role.

Acknowledgement

Special thanks to Shixun Zhang and Tingxing Dong for their guidance. Thanks to the School of Physics and Engineering of Zhengzhou University and the Supercomputing Center of Zhengzhou University for their support.

References

[1] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 task 4: Sentiment analysis in Twitter," in *Proceedings of the 11th International Workshop on Semantic Evaluation*, ser. SemEval '17. Vancouver, Canada: Association for Computational Linguistics, August 2017.

[2] M. Cliche, "Bb_twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms," *arXiv preprint arXiv:1704.06125*, 2017.

[3] H. Peng, E. Cambria, and A. Hussain, "A review of sentiment analysis research in chinese language," *Cognitive Computation*, vol. 9, no. 4, pp. 423–435, Aug 2017. [Online]. Available: https://doi.org/10.1007/s12559-017-9470-8

[4] X. Zhou, X. Wan, and J. Xiao, "Attention-based lstm network for cross-lingual sentiment classification." in *EMNLP*, 2016, pp. 247–256.

[5] C. Zhang, D. Zeng, J. Li, F.-Y. Wang, and W. Zuo, "Sentiment analysis of chinese documents: From sentence to document level," *Journal of the Association for Information Science and Technology*, vol. 60, no. 12, pp. 2474–2487, 2009.

[6] S. Tan and J. Zhang, "An empirical study of sentiment analysis for chinese documents," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2622 – 2629, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417407001534

[7] X. Zhang and Y. LeCun, "Which encoding is the best for text classification in chinese, english, japanese and korean?" *arXiv preprint arXiv:1708.02657*, 2017.

[8] Y. Lin, H. Lei, J. Wu, and X. Li, "An empirical study on sentiment classification of chinese review using word embedding," *arXiv preprint arXiv:1511.01665*, 2015.

[9] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[10] Y. Kim, "Convolutional neural networks for sentence classification," *CoRR*, vol. abs/1408.5882, 2014. [Online]. Available: http://arxiv.org/abs/1408.5882

[11] L.-W. Ku and H.-H. Chen, "Mining opinions from the web: Beyond relevance retrieval," *Journal of the Association for Information Science and Technology*, vol. 58, no. 12, pp. 1838–1850, 2007.

[12] L. Abdi and S. Hashemi, "To combat multi-class imbalanced problems by means of over-sampling techniques," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 1, pp. 238–251, 2016.

[13] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, pp. 1119–1130, Aug 2012.

[14] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[15] R. Barandela, R. Valdovinos, J. Sanchez, and F. Ferri, "The imbalanced training sample problem: Under or over sampling?" *Structural, syntactic, and statistical pattern recognition*, pp. 806–814, 2004.

[16] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, "Dimensional sentiment analysis using a regional cnn-lstm model," in *ACL 2016—Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin, Germany*, vol. 2, 2016, pp. 225–230.

[17] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Convolutional neural network committees for handwritten character classification," in *2011 International Conference on Document Analysis and Recognition*, Sept 2011, pp. 1135–1139.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'12. USA: Curran Associates Inc., 2012, pp. 1097–1105. [Online]. Available: http://dl.acm.org/citation.cfm?id=2999134.2999257

[19] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," 1999.

[20] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5, pp. 602 – 610, 2005, iJCNN 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608005001206

[21] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *CoRR*, vol. abs/1409.1259, 2014. [Online]. Available: http://arxiv.org/abs/1409.1259

[22] C. Baziotis, N. Pelekis, and C. Doulkeridis, "Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017, pp. 747–754. [Online]. Available: http://www.aclweb.org/anthology/S17-2126

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[24] T. G. Dietterich *et al.*, "Ensemble methods in machine learning," *Multiple classifier systems*, vol. 1857, pp. 1–15, 2000.