A BERT-Based Ensemble Learning Approach for Sentiment Classification in Twitter*

Shupei Li[1] and Ziyi Xu[1]

LIACS, Leiden University, Leiden, Netherlands
{s3430863, s3649024}@umail.leidenuniv.nl

Abstract. To be continued.

Keywords: Sentiment Analysis · BERT · Ensemble Learning

1 Introduction

Sentiment analysis, a growing field today, is the process of analyzing pieces of writing to determine the emotional tone they carry. In simple words, sentiment analysis helps to find the author's attitude. This method can be used by businesses to analyze product reviews and feedback, especially for social media companies with large information streams because of the wealth of data they generate. Researchers also have a special interest in social media data because of their easy availability and rapid change.

SemEval is an International Workshop on Semantic Evaluation, formerly SensEval. It is an ongoing series of evaluations of computational semantic analysis systems. The SemEval-2017 Task 4 focuses on classifying and quantifying the sentiment of tweets. The task was included in this workshop in the previous year [3], and Sentiment Analysis in Twitter has been run yearly since 2013 [4]. The subtask A of this task is the problem we try to unravel in this article. It is about to decide the overall sentiment of tweets and marks them on a three-point ordinal scale.

To achieve a higher accuracy, we don't choose CNN, LSTM, or supervised SVM but apply the more powerful BERT. Specially, we investigate BERT and its two variants in our project. In order to leverage the advantages of different models, we design an ensemble learning approach rather than just using BERT model alone. The basic idea is dividing the classification task into two stages. In the first stage, we train a series of base models to generate corresponding predictions. These predictions are the inputs of the meta model in the second stage. The final predictions are the outputs of the meta model.

The remainder of the paper is structured as follows. Section 2 discusses previous work related to sentiment classification. Following that, Section 3 introduces the datasets provided by SemEval. We explain the working principle of our ensemble learning approach in Section 4. Section 5 describes the procedure

* Text Mining, Master CS, Fall 2022, Leiden, the Netherlands

of experiments in detail and reports the experimental results. The paper is concluded with a discussion about results and a brief summary. We also attach the contributions of group members at the end of the paper.

2 Related Work

3 Data

The dataset consists of 11 files with tweets from 2013 to 2015. Tweets are marked with sentiment labels on a three-point scale {Positive, Neutral, Negative}. Each tweet corresponds to one row in datasets following a fixed format: [`id`, `sentiment label`, `text`].

The criterion of tweet selection is covering popular topics at the time of sending tweets. Datasets are downloaded via the Twitter API and have been preprocessed by workshop in advance, where three kinds of data have been removed: repeated tweets, the bag-of-words cosine similarity exceeded 0.6, and topics with less than 100 tweets. CrowdFlower is used to create all annotations on both training set and testing set. Each tweet is annotated by at least five people to ensure the accuracy of annotations. Another main quality control measure is performing hidden tests to filter out unqualified annotations. There are also manual inspections on pilot runs aiming at adjusting the annotation instructions dynamically. Table 1 summarizes the descriptive statistics of 11 files.

Table 1. Descriptive Statistics of DataSets

Dataset	Positive	Neutral	Negative	Total
2013train	3,639	4,586	1,458	9,683
2013dev	575	738	340	1,653
2013test	1,474	1,513	559	3,546
2014sarcasm	20	7	21	48
2014test	981	669	202	1,852
2015train	170	253	65	488
2015test	1,038	987	364	2,389
2016train	3,017	2,001	849	5,867
2016dev	829	745	391	1,965
2016devtest	994	680	325	1,999
2016test	7,059	10,341	3,231	20,631

According to Table 1, 2016test file has 20,631 records in total, which is much larger than any other dataset. If we only train the model on 2016train file and test it on 2016test file, the imbalance in data volume would prevent the model from capturing enough information during training and lead to a poor generalization ability. Our solution is concatenating all the other 10 files as the training set. As a result, the training set contains 29,490 records, while the test set (2016test)

includes 20,631 records. It is worth mentioning that records in 2016test file have a redundant `\t` before `\n`. We remove the unnecessary `\t` to ensure the dataset would be read into Python appropriately.

4 Methodology

4.1 BERT

BERT [1] is a popular NLP model that has achieved the remarkble performance on various tasks, such as text classification, question answering, etc. Compared to traditional RNN models, it encodes sequences in both directions instead of following a left-to-right or right-to-left routine, which is more closer to how humans understand the meaning of the text. Its bidirectional encoding ability is accomplished by the transformer architecture, which is a stack of encoders using multi-head attention mechanism. Specifically, encoders process the entire sequence at once and use layer-wise tensor operations to learn relationships between words in a sentence. This design not only encodes the inputs bidirectionally but also eliminates the possible local bias, for it gives equal importance to the local context and the long-distance context. It is worth mentioning that the training process of BERT is more efficient than RNN due to the feasibility of parallelization.

BERT requires a special format of input called WordPiece. The WordPiece tokenizer splits words into tokens and adds special tokens at the beginning as well as the end of the sentence. Preprocessed inputs provide three aspects of information for BERT model: tokens, sentence segments, and the absolute position. We perform the tokenization operation on both the training set and the test set before modeling.

We also consider two variants of BERT in our project – RoBERTa [2] and DistilBERT [5]. RoBERTa is an optimized version of BERT model. Authors of RoBERTa find that the original BERT is actually undertrained after reproducing BERT experiments. Their solution is increasing the training epochs of BERT and carefully select hyperparameters, which significantly improves the model performance in practice. On the contrary, DistilBERT is a compressed version of BERT. The main idea of DistilBERT is reducing the model size via distillation technique. Distillation consists of a teacher model and a student model. The teacher model is trained on a large dataset and is fine-tuned to maximize the accuracy. However, many features learned by the teacher model are redundant for a specific task. Therefore, we can train a much smaller student model to focus on key features and imitate the output of the teacher model. Experimental results show that DistilBERT is cheaper to train while maintaining a comparable performance to BERT.

4.2 Ensemble Learning

Ensemble learning refers to methods that combine multiple models to achieve better performance in machine learning. It encourages base models to learn different aspects of the data to reduce errors and avoid being entrapped in local

optima. In the project, sentiment analysis in Twitter is a multi-class classification problem. And we develop a classification voting ensemble model integrated BERT and its variants. Figure 1 illustrates the architecture of our model.

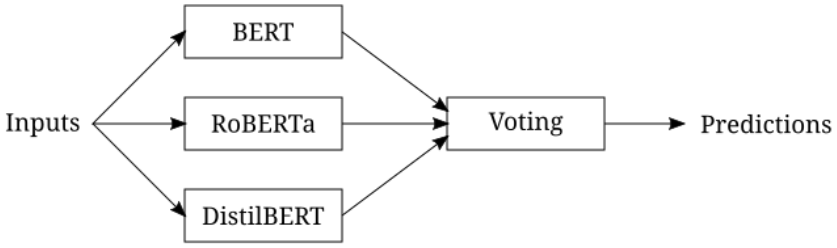

Fig. 1. The Architecture of the Proposed Model

Each base model outputs the class prediction for each record. Then, the voting classifier adopts the plurality voting strategy to generate the final prediction. In other words, the final prediction is the class label received the majority votes from base models. If all classes have the same votes, the voting classifier will choose a class label randomly as the output.

5 Experiments

5.1 Experimental Setup

Software, hardware, baselines, metrics.

5.2 Results

6 Discussion

7 Conclusion

8 Contributions

References

1. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. ArXiv **abs/1810.04805** (2019)
2. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. ArXiv **abs/1907.11692** (2019)

3. Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., Stoyanov, V.: SemEval-2016 task 4: Sentiment analysis in Twitter. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016). pp. 1–18. Association for Computational Linguistics, San Diego, California (Jun 2016)
4. Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., Wilson, T.: SemEval-2013 task 2: Sentiment analysis in Twitter. In: Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). pp. 312–320. Association for Computational Linguistics, Atlanta, Georgia, USA (Jun 2013)
5. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. ArXiv **abs/1910.01108** (2019)