Ferryman at SemEval-2020 Task 7: Ensemble Model for Assessing Humor in Edited News Headlines

Weilong Chen, Jipeng Li, Chenghao Huang, Wei Bai, Yanru Zhang, and Yan Wang*
University of Electronic Science and Technology of China
chenweilong1995,jeeperly,huangchenghao,cellur@std.uestc.edu.cn
yanruzhang,yanbo1990@uestc.edu.cn

Abstract

Natural language processing (NLP) has been applied to various fields including text classification and sentiment analysis. In the shared task of assessing the funniness of edited news headlines, which is a part of the SemEval 2020 competition, we preprocess datasets by replacing abbreviation, stemming words, then merge three models including Light Gradient Boosting Machine (LightGBM), Long Short-Term Memory (LSTM), and Bidirectional Encoder Representation from Transformer (BERT) by taking the average to perform the best. Our team Ferryman wins the 9th place in Sub-task 1 of Task 7 - Regression.

1 Introduction

With the increased influence of social media on modern society, massive news emerges on the Internet every single day. To attract people's attention, editing news headlines to make them more humorous is now an effective approach. However, considering the amount of headlines, it is impractical to manually modify each headline one by one. To solve this problem, some automatic methods to add humor in news headlines are needed. Assessing the funniness of edited news headlines plays a significant role there. Here we present the method to automatically assessing the funniness of edited news headlines using machine learning algorithms.

In SemEval-2020 Task 7: Assessing the Funniness of Edited News Headlines, nearly all existing humor datasets are annotated to study whether a short text is funny(Hossain et al., 2020). The goal of the task is to determine how machines can understand humor generated by short edits. The task has been divided into two sub-tasks: 1) given the original and edited headline, predicting the mean funniness of the edited headline; 2) given the original headline and two edited versions, predicting which edited version is the funnier of the two. The two sub-tasks are independently evaluated by root mean square error (RMSE) on the overall test set.

Since our team has participated in multiple tasks in SemEval-2020 simultaneously, we only selected the Sub-task 1 in Task 7. In this task, we encountered two main problems: (i) Training single model has no significant effect; (ii) The raw dataset fits Bidirectional Encoder Representation from Transformer (BERT) badly. To address problem (i), we adapt and fine-tune the merged model, such as increasing weights of edited words in Light Gradient Boosting Machine (LightGBM), and input the sentences before and after modification as features in Long Short-Term Memory (LSTM). Then we merge three models, which are LightGBM, LSTM and BERT respectively. To accelerate model training, we improve LightGBM with Word2Vec, FastText, and Global Vectors for Word Representation (GloVe). And to reduce , we use LSTM with max pooling and min pooling, supplementing with Enhanced Sequential Inference Model (ESIM). We also use multi-sample dropout, Frequent Pattern Growth (FPG), 5-fold cross validation, and last-5-[CLS](Cxy, 2019) to enhance the effect of BERT. Besides, we adopt some preprocessing methods such as abbreviation replacement and word stemming to achieve a better result on BERT as mentioned in problem (ii).

*All the corresponding to Yan Wang.

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 1008–1012
Barcelona, Spain (Online), December 12, 2020.

In the rest of this paper, we organize the content as follows. Related work of humor detective and the merged models will be presented in Section 2. Section 3 introduces data description, details of preprocessing, and the methodology of our models. Experimental results are discussed in Section 4. We also present the conclusion of our work at the end of paper.

2 Related Work

Humor is a complex emotion, which needs the ability to perceive fully understanding of the connection or relationship between objects in various occasions. In recent years, automatic humor recognition technology has developed rapidly. Nevertheless, to the best of our knowledge, there are not much work in this domain, because detecting humor requires a lot of cross-domain and cross-cultural knowledge, which makes it complex. For instance, a joke that is generally interesting to British may not be equally funny to Chinese because of different culture background.

On a binary humor identification task, Cattle and Ma have explored the use of semantic relatedness features based on word associations, in the meanwhile, they identify several factors that make word associations a better fit for humor (Cattle and Ma, 2018). A rich set of features for text interpretation and representation to train classification procedures in detection of irony and humor in tweet has been proposed in 2014, which achieves state-of-the-art performance in cross-domain classification experiments (Barbieri and Saggion, 2014). Yang et al. have developed a simple and effective method to extract anchors that enable humor in a sentence(Yang et al., 2015). This methodology is effective in automatically distinguishing between humorous and non-humorous texts.

There is a technique that can be implemented in the task to improve the performance. LSTM, was firstly proposed in 1991 (Hochreiter and Schmidhuber,), is an extension of recurrent neural network, which have been implemented in several Natural language processing (NLP) tasks such as sentiment classification, neural translation, and language generation etc (Liu et al., 2019). Some practical attempts have been carried out in this field. Nabil Hossain et al. have used LSTM with 16 hidden unites to detect humorous headlines (Hossain et al., 2019). Bertero and Fung show how the LSTM effectively models the setup-punchline relation reducing the number of false positives and increasing the recall (Bertero and Fung, 2016).

3 Methodology and data

3.1 Data Description

Humicroedit is a novel dataset released for the research in computational humor, which consists of English news headlines and their edited versions with simple substitute designed to make them funny. The dataset are constructed by the popular news headlines on the social media site Reddit (Hossain et al., 2019). There are 15,095 edited funny headlines and five humor scores for each one respectively from the crowdsourcing editors and judges. The funny score ranges from 0 to 3, and the overall mean funniness score of the edited headline is 0.94. The headlines have 4 words at least and the longest one have 20 words.

This humor data set is suitable for computational humor research due to the following reasons. (i) Headline do not have static pattern ,which include few words but convey abundant information. (ii) Making headlines interesting needs a deeper understanding of world-knowledge and common-sense. (iii) Humorous headlines are often generated using several layers of cognition and reasoning. As the reasons mentioned above, the artificial intelligence tools of NLP not only need to be robust at pattern recognition, but also capable of deeper semantic understanding and reasoning (Hossain et al., 2019).

3.2 Data Preprocessing

We first preprocess the data before feeding the data set to the model. In this section, we will introduce the core methods and strategies of data processing.

Abbreviation Replacement - It is widely known that people on social platforms usually use abbreviations to comment. As for the offensive tweet, euphemisms are often used to replace the original presentation. For instance, ”G9” is the abbreviation of ”good night”. In order to better understand the content, we created a dictionary to substitute abbreviations in the data set, which was an effective strategy.

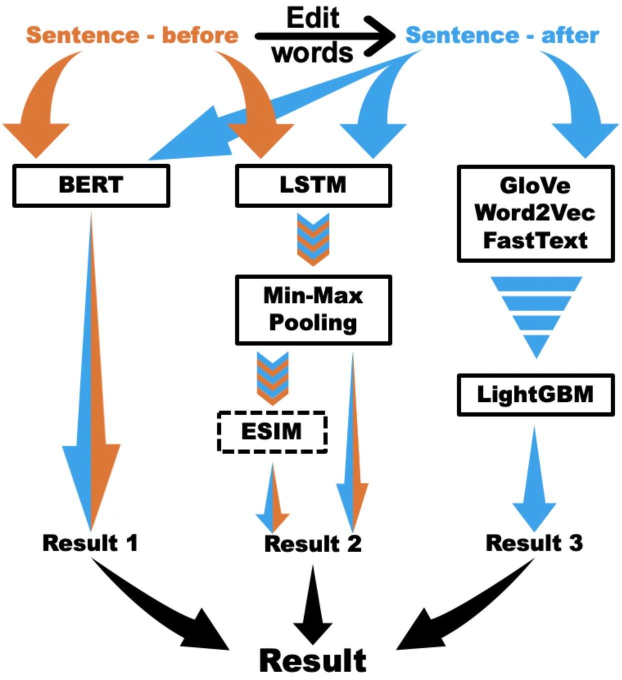

Figure 1: Experimental Process of the Task

Word Stemming - Word stemming is the approach of removing the affix to obtain the root. For example, the Word stemming operation can simplify ”stemming”, ”stemmer”, ”stemmed” to a common root ”stem”. We use this method to map related words to the same stem, which generally gives satisfactory results, even if the stem is not a valid root of the word.

Other Normalization Approaches - We convert the characters to lowercase and delete the stopwords which are meaningless and make up a great amount of the whole content.

3.3 Methodology

After data preprocessing, we transform words in corpus into vectors and train three models including LightGBM, LSTM and BERT to find the best. First of all, for BERT, we use the data before and after processing to observe the prediction results. For LSTM, we use max pooling and min pooling here, and combine with ESIM also made predictions on the data before and after processing. In LightGBM, we use GloVe for the pre-processed data. GloVe uses the frequency of word frequency co-occurrence in the corpus as the target of word vector learning approximation, and combine Word2Vec and FastText to vectorize the data as the input of LightGBM and then predict. We simply add all the three 300-dimention vector with average pooling in the LightGBM. Finally, the results of the three models are obtained, and we get the best result by comparison. We will describe the three models and parameter adjustment methods in detail below.

LightGBM - LightGBM is a new Gradient Boosting Decision Trees (GBDT) implementation with Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB)(Ke et al., 2017). Through their experiments on multiple public datasets, LightGBM speeds up the training process of conventional GBDT by up to over 20 times, while achieving almost the same accuracy. In order to make this method more suitable for our task training, we also combine other methods to learn word vectors including Word2Vec, FastText and GloVe. Word2Vec, FastText and GloVe are all different methods of generating word vectors. The word vectors generate by these methods are input into the LightGBM model, which produces the result 3.

LSTM - LSTM is a more powerful extension of recurrent neural network (RNN)(Hochreiter and Schmidhuber, 1997), which is a special RNN network designed to solve the problem of long dependency.

LSTM has a chain structure similar to RNN, but its repeating units are different from the units in a standard RNN network with only one network layer, its internal four network layers. Meanwhile, we use max pooling and min pooling to reduce parameters, control overfitting, improve model performance, and save computing power. In addition to the traditional LSTM, we also use ESIM, the improved LSTM, to process the data. ESIM is a carefully designing sequential inference models based on chain LSTMs, and also explicitly considers recursive architectures in both local inference modeling and inference composition(Chen et al., 2016). After the pooling is used as the input of ESIM, we compare this result with LSTM to get the result 2.

BERT - BERT is developed by Google research team(Devlin et al., 2018). The main innovations of the model are in the pre-train methods, Masked Levenberg-Marquard algorithm and Next Sentence Prediction, which are used to capture the representation of words and sentences respectively. BERT uses identical multi-head transformer structure(Vaswani et al., 2017), is pre-trained on huge corpus from different sources. Here, we use FPG algorithm to stored the data set with a data structure called a Frequent Pattern Tree(Han et al., 2000). For better generalization of the model, multi-sample dropout is adopted to avoid overfitting, which is an efficient regularization technique for both accelerating training and improving generalization over the original dropout(Inoue, 2019). We conduct a 5-fold cross-validation on the basis of the model, and then vote to further improve the score, and finally achieve good performance. The construction of our model is shown in Fig.1.

4 Result

RMSE@N% most/least funny headlines	Precision(Rank)
RMSE	0.52776(9)
RMSE@10	0.83953(39)
RMSE@20	0.72031(40)
RMSE@30	0.63722(41)
RMSE@40	0.57503 (40)

Table 1: The results of experiments

The results of our comprehensive model on the official test sets for SemEval Sub-task 1 are displayed in the table 1. The table reports RMSE by taking the N% most funny headlines and N% least funny headlines in the test set, for $N \in \{10, 20, 30, 40\}$. RMSE is the square of the deviation of the predicted value from the true value, and then the square root of the number of observations. Insides, the brackets is our performance ranking with other teams.

Model	RMSE
LightGBM	0.82642
LSTM	0.75671
BERT	0.63591

Table 2: The results of the three models

Among the three models, LightGBM performed best, BERT has the worst performance, while LSTM's performance is centered, as shown in the table 2. It may be because the method of generating word vectors in LightGBM can obtain text features more effectively, so that the LightGBM can play its role better. And BERT uses too many label substitutions to affect the model performance. We take the average of the predicted values after considering compromises. Our model ranks the 9th place by RMSE on the overall test set.

5 Conclusion

In this work, we have presented our system for SemEval Sub-task 1. In Sub-task 1, we have trained three models by fine-tuning the parameters and preprocessing the dataset to detect humor in news headlines.

After comparing three methods, LightGBM performs best on the RMSE while BERT does the opposite, and LSTM is somewhere between the two. Considering a compromise, we merge them by taking the average of their predicted values.

To conclude, the evaluation results indicate that our system is capable of detecting humor in news headlines robustly. However, how to tune the parameters is nontrivial, and there are more efficient ways to be explored, which could yield better performance, especially on BERT. In the future, in order to increase the accuracy of the system, we plan to improve the performance of BERT by taking more preprocessing mechanisms and deeper feature extraction on edited words, and fine-tuning parameters further.

References

Francesco Barbieri and Horacio Saggion. 2014. Automatic detection of irony and humour in twitter. *Process Biochemistry*, 40(8):2637–2642.

Dario Bertero and Pascale Fung. 2016. A long short-term memory framework for predicting humor in dialogues. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Andrew Cattle and Xiaojuan Ma. 2018. Recognizing humour using word associations and humour anchor extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1849–1858, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2016. Enhanced lstm for natural language inference. *arXiv preprint arXiv:1609.06038*.

Xs Cxy. 2019. Sentiment analysis of internet news.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jiawei Han, Jian Pei, and Yiwen Yin. 2000. Mining frequent patterns without candidate generation. *ACM sigmod record*, 29(2):1–12.

S Hochreiter and J Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Nabil Hossain, John Krumm, and Michael Gamon. 2019. “president vows to cut <taxes> hair”: Dataset and analysis of creative text editing for humorous headlines. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 133–142, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Nabil Hossain, John Krumm, Michael Gamon, and Henry Kautz. 2020. Semeval-2020 Task 7: Assessing humor in edited news headlines. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2020)*, Barcelona, Spain.

Hiroshi Inoue. 2019. Multi-sample dropout for accelerated training and better generalization. *arXiv preprint arXiv:1905.09788*.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*, pages 3146–3154.

Ping Liu, Wen Li, and Liang Zou. 2019. NULI at SemEval-2019 task 6: Transfer learning for offensive language detection using bidirectional transformers. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 87–91, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Diyi Yang, Alon Lavie, Chris Dyer, and Eduard Hovy. 2015. Humor recognition and humor anchor extraction. In *Conference on Empirical Methods in Natural Language Processing*.