Gulu at SemEval-2021 Task 7: Detecting and Rating Humor and Offense

Maoqin Yang
School of Information, Yunnan University, Yunnan, P.R. China
maomaoq33@gmail.com

Abstract

Humor recognition is a challenging task in natural language processing. This document presents my approaches to detect and rate humor and offense from the given English text. This task includes 2 tasks: task 1 which contains 3 subtasks (1a, 1b, and 1c), and task 2. Subtask 1a and 1c can be regarded as classification problems and take ALBERT as the basic model. Subtask 1b and 2 can be viewed as regression issues and take RoBERTa as the basic model. And finally, team-Gulu scores in subtask 1a with a weighted average F1 score of 0.9190, in subtask 1b with an RMSE score of 0.7405, in subtask 1c with a weighted average F1 score of 0.5561, and in subtask 2 with an RMSE score of 0.5807 on the private leader board.

1 Introduction

For social animals like humans, humor is an effective bonus. From the perspective of evolutionary psychology, "humorous" often means superior creativity, in other words, a smarter mind. Therefore, it is important to recognize whether a sentence is humorous and how humorous the sentence is. It is a bit impractical to recognize such a huge data set by humans, so it becomes necessary for us to develop a system to automatically detect humor. In this task, the organizer collects labels and ratings from a balanced age group of 18-70. Annotators also represent various genders, political positions, and income levels. Therefore, for some texts classified as humorous, we should once again prove whether they are controversial and predict the offensiveness of the text. For more specific content, please refer to the official website of the competition [1].

[1] https://competitions.codalab.org/competitions/27446

Because the pre-trained and deep learning models have shown excellent performance in many NLP problems such as classification and topic extraction(Zampieri et al., 2019), so I use deep learning methods to deal with those four tasks. According to the latest related research progress, the transformer-based language model has become my favorite model. In order to facilitate the understanding of the corresponding model of each subtask, I made it into a table shown as Table 1. I choose A Lite BERT (ALBERT)(Lan et al., 2019) as my basic model in subtask 1a. In subtask 1b and subtask 2, I choose Bidirectional Encoder Representations for Transformers (BERT)(Devlin et al., 2018) model as my basic model. In subtask 1c, A Robustly Optimized BERT (RoBERTa)(Liu et al., 2019) has been chosen. To get a more effective and higher accuracy model in subtask 1a, BiGRU combined with attention. To prove the effectiveness of this model, there are also comparative experiments with other neural networks for task 1c. To obtain as much effective information as possible from the limited data, the 5-fold cross-validation method has been used.

2 Related Work

Automatic humor recognition is a very challenging research topic in natural language processing. A person's degree of humor is largely determined by his educational knowledge and common sense of life. In addition, many types of humor require a lot of external knowledge, such as irony, metaphor and satire.

Yang et al. (2015) first determined the semantic structure behind each structure of the humor and design feature set, and then used a calculation method to identify the humor and their humor recognizer was very effective in automatic distinction humorous and non-humorous text.

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 1120–1124
Bangkok, Thailand (online), August 5–6, 2021.

subtask	1a	1b	1c	2
category	classification	reg	classification	reg
model	ALBERT+BiGRU	BERT	RoBERTa+BiLSTM+BiGRU	BERT

Table 1: The category and model used for each subtask, where the reg stands for $regression$

Morales and Zhai (2017) proposed a generative language model based on the inconsistency theory to model humorous text, so that they can use background text sources such as Wikipedia item descriptions, and can build multiple functions for recognizing humorous comments. Using supervised learning to classify reviews into humorous reviews and non-humorous reviews, these functions showed that the features constructed based on the proposed generative model were more effective than the main features proposed in the existing literature.

Liu et al. (2018) found that certain grammatical structural features are consistently related to humor. Both experimental results and analysis showed that humor can be regarded as a style, and the content-independent syntactic structure can help identify humor and had good explanatory power. Therefore, they proposed to use syntactic structure features to enhance humor recognition ability. Compared with the baseline driven by humor theory, their method had achieved a significant improvement.

And subtask 1b and 2 are regression problems. We need to predict the humor of a sentence, and because the implicit meaning of the sentence may be offensive for someone, we also need to detect the degree of attack on each sentence. Traditionally, text regression is solved using linear models. Bitvai and Cohn (2013) proposed a method based on a deep convolutional neural network (CNN). Yang et al. (2015) recommended using copula: a powerful statistical framework. Their model clearly outperformed a strong linear and nonlinear discrimination baseline. Subramanian et al. (2018) used CNN regression with auxiliary ordinal regression objective to predict the popularity of petitions in their work. A regression task is actually a special form of the classification task. The final output is a value rather than the probability of a specific category. Therefore, the BERT model can achieve good results.

3 Materials and Methods

3.1 Preprocessing

The given data(Meaney et al., 2021) contains the tasks required by each subtask, but some corresponding data are missing. For example, if a sentence is judged as not humorous in 1a, there will be no data in the column (the fourth and fifth column) corresponding to subtask 1b and 1c. To facilitate the experiment, I added 0 to all missing values.

3.2 Data set

Given a sentence, for 1a, the system must assign the label to 1 if it is recognized as is_humor, otherwise, assign it to 0. And if there is a $humor\_controversy$ in 1c, the corresponding label is 1. For this task, the available sentences including 6948 training sentences, 1052 development sentences, and 1000 testing sentences. The label distribution in the training set is almost balanced for 1c, but for 1a, label 1 accounts for only 38.1% of the total after assigning all missing values to 0. The number of sentences for each label is listed in Table 2.

task	label 0	label 1
subtask 1a	2652	4296
subtask 1c	2149	2147

Table 2: The distribution of training set

I divide all the data of subtask 1b and 2 into 5 intervals (take 1 as the step size) and count the total of each interval. It can be seen from Figure 1 that most of the data of $humor\_rating$ are between 1 and 2 (1361 sentences), and there is no data between 4 and 5. The label $offense\_rating$ scores of 0 accounted for the majority (2913 data in total).

3.3 Classification

Text classification is the most basic and very necessary task in natural language processing (NLP). Two of this task belongs to classification problems. In subtask 1a, I combine ALBERT with BiGRU-Attention. In subtask 1c, I combined RoBERTa with BiLSTM+BiGRU.

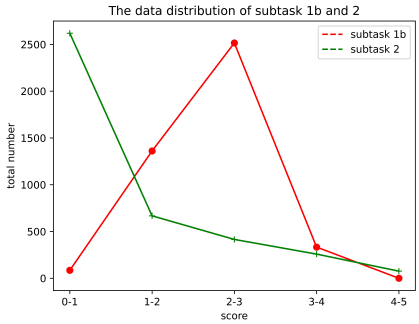

Figure 1: The data distribution of subtask 1b and 2

3.3.1 ALBERT + BiGRU-Attention

The ALBERT model is an improvement based on the BERT model. The ALBERT model[2] has designed parameter reduction by changing the result of the original embedding parameter P (the product of the vocabulary size V and the hidden layer size H).

$$V * H = P \rightarrow V * E + E * H = P \quad (1)$$

Where E represents the size of the low-dimensional embedding space. In ALBERT, $H >> E$. The self-supervised loss is used to focus on the internal coherence in the construction of sentences[3].

The BiGRU-Attention model[4] is divided into three parts: text vector input layer, hidden layer, and output layer. Among them, the hidden layer consists of three layers: the BiGRU layer, the attention layer, and the Dense layer (fully connected layer). The output of the ALBERT model will be used as the input. After receiving the input, the BiGRU neural network layer will extract features of the deep-level information of the text firstly. Secondly, it uses the attention layer to assign corresponding weights to the deep-level information of the extracted text. Finally, the text feature information with different weights is put into the softmax function layer for classification.

In order to improve the classification ability of the model, I combined ALBERT and BiGRU-Attention. The model diagram is shown in Figure 2.

[2] https://huggingface.co/albert-base-v2

[3] https://zhuanlan.zhihu.com/p/162275803

[4] https://blog.csdn.net/qq_40900196/article/details/88998290

3.3.2 RoBERTa + BiLSTM + BiGRU

RoBERTa[5] mainly made several adjustments based on BERT: 1) Longer training time, larger batch size, more training data; 2) Removed next predict loss; 3) Longer training sequence; 4) Dynamic adjustment Masking mechanism.

Using the BiLSTM model can better capture the two-way semantic dependence. Because LSTM can learn what information to remember and what information to forget during the training process. BiGRU is a unidirectional, opposite direction, and outputs a neural network model composed of GRUs determined by the states of these two GRUs. At each moment, the input will provide two GRUs in opposite directions at the same time, and the output will be jointly determined by the two unidirectional GRUs.

In order to improve the ability of the model, I combined RoBERTa and BiLSTM+BiGRU. The model diagram is shown in Figure 3.

3.4 Regression

What regression predictive modeling needs to accomplish is to approximate a mapping function from an input variable to a continuous output variable. Both regression subtasks use the BERT model.

The BERT model implements three embedding layers: position embedding, word embedding, and segment embedding. BERT uses two training strategies: the masked language model and the next sentence prediction. The language model trained in this way usually has a deeper sense of language context and can be further applied to process various NLP tasks(classification et.), with an additional output layer(Fan et al., 2019).

3.5 Evaluation

The main metric for the classification tasks will be f1-measure(wei et al., 2020).

$$precision = \frac{true \quad positive}{true\, positive + false\, positive} \quad (2)$$

$$recall = \frac{true \quad positive}{true\, positive + false\, negative} \quad (3)$$

$$F1 = \frac{2 * precision * recall}{precision + recall} \quad (4)$$

[5] https://huggingface.co/roberta-base

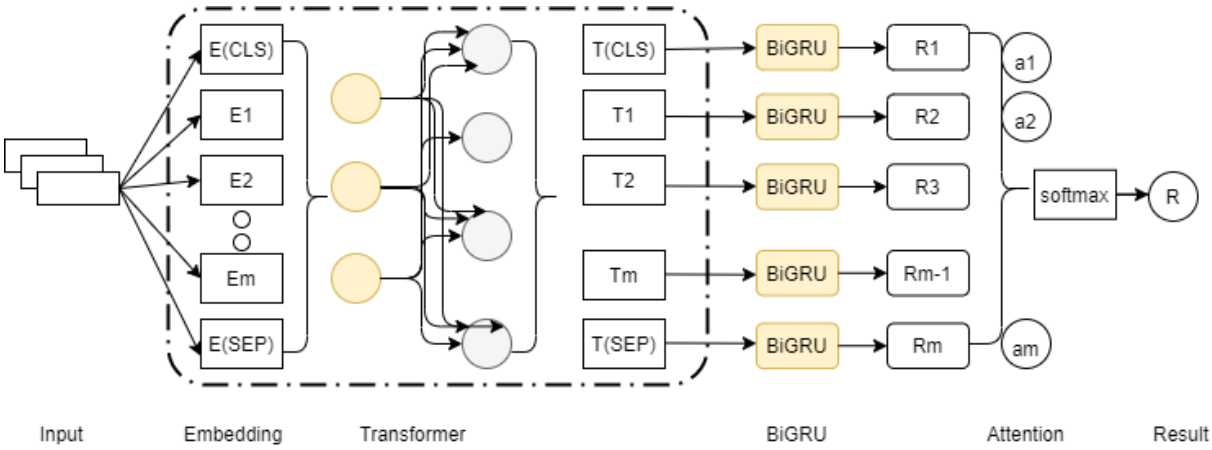

Figure 2: ALBERT+BiGRU-Attention for task 1a, where the $E[CLS]$ and $E[SEP]$ are added at the beginning and end of each instance respectively

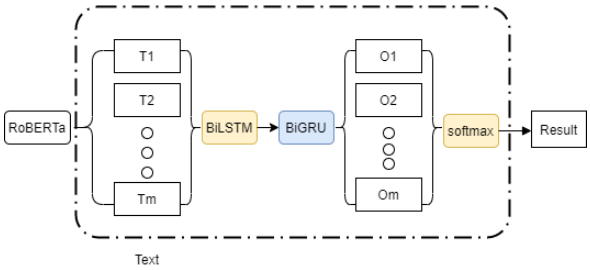

Figure 3: The model for task 1c

Model	step	batch size	lr	epoch
BERT	500	32	2e-5	2
ALBERT	2500	32	2e-5	10
RoBERTa	5000	16	2e-5	10

Table 3: The parameters, where the lr stands for $learningrate$

The metric for the regression tasks will be the root mean squared error (RMSE). Below x and y are D dimensional vectors, and x_i represents the value of x in the ith dimension.

$$RMSE = \sqrt{\sum_{i=1}^{D}(x_i - y_i)^2} \quad (5)$$

4 Results

In this task, I used ALBERT, RoBERTa, and BERT models for the training task. For these models, the main hyperparameters I want to pay attention to are the training step size, batch size learning rate, and epoch. The parameters of my model are shown in Table 3.

4.1 Classification results

For task 1c, several sets of comparative experiments were carried out. The comparison results are listed in Table 4, and the cross-validation results are 0.92, 0.94, 0.94, 0.93, and 0.67 respectively.

All results are the results of the evaluation set. The output of the classification result is shown in Table 5. We can see that the number of label 1 is close to twice the number of label 0.

Task 1a scores 0.9190 but 1c scores 0.5561. From the distribution of their data, this may be because the data of task 1c is not balanced. Moreover, because task 1c has a dependency on task 1a, only filling in missing values with 0 may affect the judgment of the system.

Model	F1-Score
RoBERTa	0.80
RoBERTa+BiGRU	0.86
RoBERTa+BiGRU+BiLSTM	0.88

Table 4: The comparative results of task 1c, and the model is the $base$ version.

task	label 0	label 1
subtask 1a	386	614
subtask 1c	331	669

Table 5: The result distribution of task 1b and 2

4.2 Regression results

The output of the regression result is shown in Figure 4. In subtask 1b, all predicted values are between 0 and 3. Most of the values are in the range

of 2 to 3 (412 sentences). In subtask 2, all predicted values are also between 0 and 3 (442 sentences score 0). Comparing with the data distribution in the training set, we find that the distribution of the predicted value is consistent with the distribution of the training data.

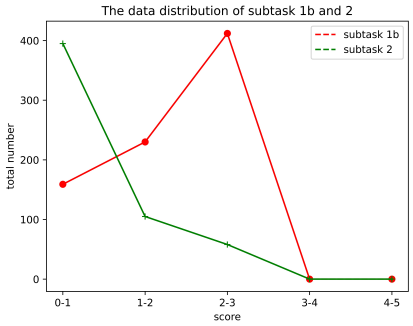

Figure 4: The predicting result of *humor_rating*

5 Conclusion and Future Work

In this work, I present my result on HaHackathon: Detecting and Rating Humor and Offense which includes four subtasks. For tasks 1a and 1c, I use the BiGRU-Attention based on the ALBERT model to complete subtask 1a, and 1c is completed by RoBERTa combine with BiLSTM+BiGRU and this model works well. I also summarized the possible reasons for the low score in task 1c.

From a theoretical and computational point of view, it is difficult to establish a mechanism for computers to understand humor like humans. The reason is as follows. 1) The definition of humor is loose. It is almost impossible to identify humor by establishing rules. 2) Humor is related to context and background. Humor expects to break the common sense of readers in a specific situation. In the future, we should design features that are interpretable, calculable, and easy to implement that conform to humor theory.

References

Zsolt Bitvai and Trevor Cohn. 2013. Non-linear text regression with a deep convolutional neural network. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics:180–185.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.

Yadan Fan, Sicheng Zhou, Yifan Li, and Rui Zhang. 2019. Deep learning approaches for extracting adverse events and indications of dietary supplements from clinical text. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers).

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, and Piyush Sharma. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. arXiv:1909.11942. Version 6.

Lizhen Liu, Donghai Zhang, and Wei Song. 2018. Exploiting syntactic structures for humor recognition. Proceedings of the 27th International Conference on Computational Linguistics:1875–1883.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, and Mandar Joshi. 2019. RoBERTa: A robustly optimized BERT pretraining approach. arXiv:1907.11692. Version 1.

J.A. Meaney, Steven R. Wilson, Luis Chiruzzo, Adam Lopez, and Walid Magdy. 2021. Semeval 2021 task 7: Hahackathon, detecting and rating humor and offense. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.

Alex Morales and Chengxiang Zhai. 2017. Identifying humor in reviews using background text sources. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing.

Shivashankar Subramanian, Timothy Baldwin, and Trevor Cohn. 2018. Content-based popularity prediction of online petitions using a deep regression model. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers):182–188.

Qiang wei, Zongcheng Ji, and zhiheng Li. 2020. A study of deep learning approaches for medication and adverse drug event extraction from clinical text. *Journal of the American Medical Informatics Association, 27(1), 2020, 13–21*.

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

M. Zampieri, S. Malmasi, P. Nakov, and S. Rosenthal. 2019. Predicting the type and target of offensive posts in social media. arXiv:1902.09666.