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

Renyuan Liu
Yunnan University, Yunnan, P.R. China
bluewind159@qq.com

Xiaobing Zhou *
Yunnan University, Yunnan, P.R. China
zhouxb@ynu.edu.com

Abstract

This paper introduces the result of Team Grenzlinie's experiment in SemEval-2021 task 7: HaHackathon: Detecting and Rating Humor and Offense in English. This task has two subtasks. Subtask1 includes the humor detection task, the humor rating prediction task, and the humor controversy detection task. Subtask2 is an offensive rating prediction task. Detection task is a binary classification task, and the rating prediction task is a regression task between 0 to 5. 0 means the task is not humorous or not offensive, 5 means the task is very humorous or very offensive. For all the tasks, this paper chooses RoBERTa as the pre-trained model. In classification tasks, Bi-LSTM and adversarial training are adopted. In the regression task, the Bi-LSTM is also adopted. And then we propose a new approach named compare method. Finally, our system achieves an F1-score of 95.05% in the humor detection task, F1-score of 61.74% in the humor controversy detection task, 0.6143 RMSE in humor rating task, 0.4761 RMSE in the offensive rating task on the test datasets.

1 Introduction

Humorous is one kind most interesting, most has the power, most has the universal significance transmission art. Therefore, humor is one of the ways to improve the quality of daily conversation. In the field of natural language processing, how to make the computer learn humor and improve the quality of human-computer interaction is an important problem. The previous researches task was only to input the humorous corpus into the deep learning network and let the algorithm learn how to generate humorous dialogue. In this case, the sentences are often problematic. Because humor is an abstract concept, in different situations, the degree of humor and the way of humor will be different. Therefore, before the computer learns to generate humorous sentences, it is an important task for the computer to understand humor and distinguish different degrees and forms of humor.

This paper mainly discusses how to identify these humorous sentences automatically. In SemEval-2021 task 7, subtask1 includes the humor detection task, the humor rating predicts task and the humor controversy detection task (Meaney et al., 2021). Subtask2 is an offensive rating predict task. In the detection task, the Bi-LSTM and adversarial training (Tramèr et al., 2017) is adopted, we also try to use FocalLoss to solve the data unbalance problem. In the regression task, the Bi-LSTM is also adopted. And then we propose a new method named compare method is also adopted.

The rest of the paper is as follows: Section 2 briefly introduces the related work. Section 3 describes the optimization approach to be used in detail. Section 4 describes the experiment process in detail. Section 5 is the conclusion of this paper.

2 Related Work

On large corpora, pre-trained models (PTMs) can learn common language representation, which is beneficial for subsequent NLP tasks and can avoid training new models from scratch (Wang et al., 2018). With the development of computing power and the improvement of training skills, the architecture of PTMs is advancing from shallow to deep.

The goal of the first version of PTMs is to learn good word embedding. Since these models are no longer needed by downstream tasks, they are usually very superficial for computational efficiencies, such as skip-gram (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). Although these pre-trained embeddings can capture the semantic meaning of words, they are context-free and cannot capture the advanced concepts in the context, such as polysemy disambiguation, syntactic structure,

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 281–285
Bangkok, Thailand (online), August 5–6, 2021. ©2021 Association for Computational Linguistics

semantic role, anaphora, and so on. The second version of PTMs mainly learn context word embedding, such as ELMo(Peters et al., 2018), OpenAI, GPT(Radford et al., 2019) and BERT(Devlin et al., 2018). These learned coders still need to represent words in context through downstream tasks. Besides, various pre-training tasks are proposed to learn PTMs for different purposes.

Given the pre-trained model, downstream algorithms like Random Forest, SVM, Logistic Regression, or single linear layer can be adopted to get the result.

Recent studies have extended the humor detection task to the field of multi-modality and used gesture, speech prosody, and other features in humor detection task(Li et al., 2020). They also began to work on the joint training and integration of humor detection task and humor generation task(Weller et al., 2020). However, these models still regard humor detection as a binary classification task, without considering humor scoring and controversy.

3 Methods Description

Baseline Model In these tasks, RoBERTa is adopted as the baseline model, and softmax is adopted as the activation function in the classification task. In the regression task first, we try to use ReLU as the activation function, then we use sigmoid as the activation function, and multiply the output of sigmoid by 5. But we find the method without activation function does the best in regression task. If we add ReLU after regression task, the RMSE will reduce 0.1 0.2. So the activation function is not adopted in the baseline model. To avoid the negative output of the model in the scoring model, we need to add ReLU as an activation function in the test phase.

In the regression task, the Mean Square Loss is adopted as the loss function. In classification tasks, we use the CrossEntropy Loss as the loss function.

Method1: Bi-LSTM In this paper, all the subtask use the Bi-LSTM to extract more abundant features. In this model, Bi-LSTM is added after the pre-trained model. [CLS] (classification symbol) always be added before sentence, and use classifier to compute [CLS] representation to get the result. So, there is a problem, that is, the sentence representation from Bi-LSTM will not integrate on symbol [CLS]. But we need the representation of [CLS] for the next step. So the output of Bi-LSTM is sent into a new defined transformer layer, encode the sentence representation into the symbol [CLS]. Finally, the sentence representation will send into a single linear layer to get the result.

Method2: adversarial training Then the adversarial training is adopted to improve the baseline model. Adversarial training is an important way to enhance the robustness of neural networks. In the process of confrontation training, the samples will be mixed with some small disturbances, and then make the neural network adapts to this change, so it has the robustness to the confrontation samples. In the field of the language model, adversarial training improves both robustness and generalization (Morris et al., 2020).

Adversarial training can be summarized as the following maximum and minimum formula,

$$\underset{\theta}{min}\,\mathbb{E}_{(Z,y)\sim\mathfrak{D}}\left[\underset{\|\theta\leq\epsilon\|}{max}(L(F_\theta(X+\delta)),y)\right] \quad (1)$$

Where X represents the input representation of the sample, θ represents the disturbance superimposed on the input, $F_\theta()$ is the neural network function, y is the label of the sample, and $L(F_\theta(X+\delta)),y)$ represents the loss obtained by superimposing a disturbance θ on the sample X, and then comparing it with the label y through the neural network function. $max(L)$ is the optimization objective, that is to find the disturbance that maximizes the loss function. In short, the added disturbance should confuse the neural network as much as possible.

$min_\theta\,\mathbb{E}_{(Z,y)\sim\mathfrak{D}}$ is the minimization formula to optimize the neural network, that is, when the disturbance is fixed, we train the neural network model to minimize the loss of training data, that is to say, the model has certain robustness and can adapt to the disturbance.

In this method, FGM (Fast Gradient Method) (Miyato et al., 2016) is adopted. The idea is very simple, that is, let the direction of disturbance increase along the gradient, and the increase along the gradient means the maximum loss. The formula of FGM is as follows.

$$\delta = \epsilon \cdot \frac{g}{\|g\|_2} \quad (2)$$

Where ϵ is a constant, which controls the degree of disturbance rejection. $g = \nabla X(L(F_\theta(X)),y)$, i.e. the gradient of loss function L with respect to input X.

Method3: FocalLoss In the classification task, we can see that the data ratio of humorous and non-humorous sentences is close to 2:1, and then in humor's data, the ratio of controversy and non-controversy is close to 1:1, but We assume that non-humorous sentences are also non-controversy sentences. Therefore, these two tasks are faced with the problem of data imbalance. To solve this problem, we use the FocalLoss (Lin et al., 2017) as the loss function. The FocalLoss is as follows.

$$FocalLoss(p_t) = -\alpha_t \, (1 - p_t \,)^{\gamma} \;\; log(\, p_t) \quad (3)$$

Where p_t is the probability of the label t that is outputted by the classifier. N is the number of labels. α and γ are constant.

Method4: Compare Method This method is only for humor rating predict task and offensive rating predict task. In the few-shot classification tasks, traditional approach is given a pair of sentences in the same class and given a pair of sentences in different classes. Let the classifier identify whether the pair of sentences is the same class or different classes. So the latent feature of each label in the sentence can be extracted. Unfortunately, this approach can't be used in regression tasks.

Based on this idea, we proposed the compare method. This method extends the above idea to the regression task. The approach is shown in figure 1. In this model, we input sentence A with rating L(A), and sentence B with rating L(B), then three different models that realize the function of M_{add} (L(A) plus L(B)), M_{sub_AB} (L(A) minus L(B)), and M_{sub_BA} (L(B) minus L(A)). It means to use these models to encode pairs of sentences and output $Z_{add}, Z_{sub\_BA}, Z_{sub\_AB}$. Then put these features into the classifier. Let the output ratings become the addition and the subtraction of the pair of sentences' rating.

Furthermore, the sentence representation which rating is close to the addition and subtraction of the pair of sentences' ratings can be used to introduce the $Z_{add}, Z_{sub\_BA}, Z_{sub\_AB}$ by minimizing the MSELoss of $Z_{add}, Z_{sub\_BA}, Z_{sub\_AB}$ and sentence representation. In this task, this approach is not adopted because of the lack of data.

The three models have the same construction. To simplify the computation, the last layer's hidden output from RoBERTa is set as the feature of each token. Then these token features are concatenated like "[CLS] (sentence) [SEP] (another sentence) [SEP]". And send the concatenated output to a single transformer layer to get the [CLS] output for classifying.

The loss function C_{loss} and the Add_{loss} is as follows. In the equation, ML is MSELoss and C is our model, $C(F_A)$ means the output of our model. These loss functions are the loss of the single sentence result and the loss of the result of $Z_{add}, Z_{sub\_BA}, Z_{sub\_AB}$.

$$\begin{aligned} C_{\text{loss}} = ML(C(F_A), L(A)) + \\ ML(C(F_B), L(B)) \end{aligned} \quad (4)$$

$$\begin{aligned} &Add_{loss} = \\ &\quad ML(C(Z_{add}), L(A) + L(B)) + \\ &\quad ML(C(Z_{sub\_AB}), L(A) - L(B)) + \\ &\quad ML(C(Z_{sub\_BA}), L(B) - L(A)) \end{aligned} \quad (5)$$

4 Experiment Setup

Datasets First of all, we try to find the relationship between tasks. In the beginning, we think that those with low humor ratings or high offensive ratings may be controversial, but unfortunately, we find many Counterexamples in the datasets. Then we tried to train several tasks together, but the result was not as good as that of training it independently. So we train these tasks independently.

Secondly, in the task of humor scoring and humor controversy detection, only humorous sentences need to be rating predicted and detected. In the data set, only humorous sentences have humor ratings and humor controversy labels. Therefore, how to deal with the label of non-humorous sentences is an important problem. We have tried to set the controversy label of non-humorous sentences to 2, that is, the third category, but this approach will identify humorous sentences as the third category, which will interfere with the model. Therefore, in this paper, we set the rating of non-humorous sentences to 0, and the controversy label to 0, i.e. non-controversy.

Parameters setting In this section, the hyper-parameter is the same in all subtasks. The optimizer is AdamW with a 3e-5 learning rate and 1e-8 adam epsilon. The pre-trained model has 12 transformer layers and 768 hidden sizes. The max sequence length is 180. The batch size is 8. And weight decay is 0.

5 Result

The result of the test datasets is shown in Table 1. Final results in line 1 is results in evaluation

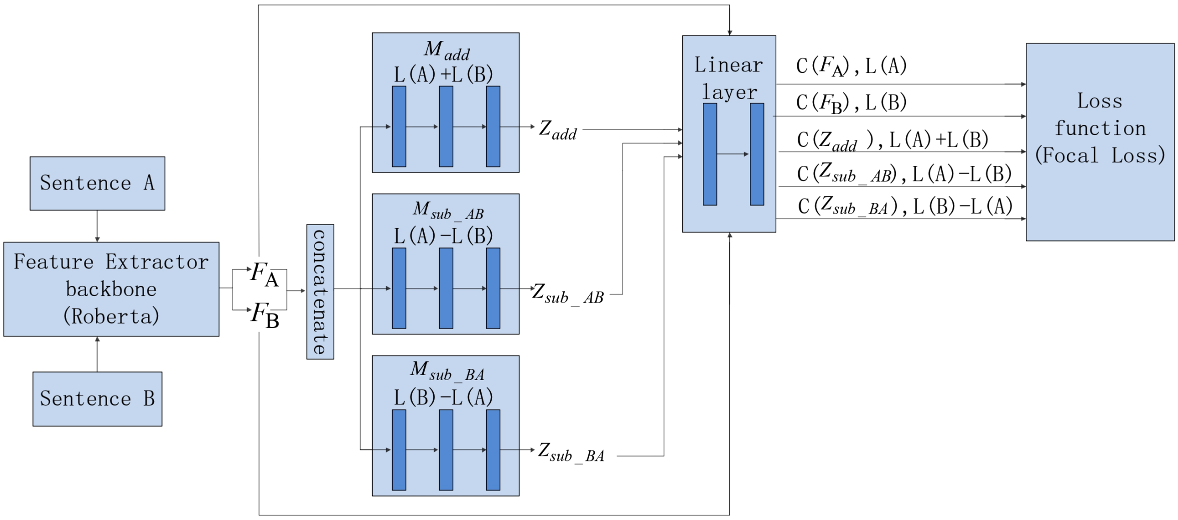

Figure 1: Compare Method Construction

phase. FGM+FocalLoss and compared method are adopted. and other line are the result in post evaluation phase. In this phase, we reduce the learning rate. In table 1, it can be seen that the result of all optimize methods in the controversy detection task is worse than the baseline model. Because we set the non-humorous sentences as non-controversy, This will greatly interfere with the model's judgment of non-controversy sentences. In the evaluation datasets, i.e. predicted results from non-humorous sentences are used to calculate the F1-score, these approaches do optimize the baseline model. But these approaches do not play an optimization role in the test phase. So we can make this conclusion. Then FGM and Bi-LSTM will make the model extract more abundant features, which will undoubtedly aggravate the interference of non-humorous sentences and reduce the prediction accuracy of the model.

FocalLoss didn't work as expected and didn't get better results. Because FocalLoss usually use in the datasets that 0 label is more than 1 label, but in the humor detection task, 1 label is more than 0. Although we adjusted the alpha in FocalLoss to 0.67, FocalLoss still failed to get better results.

FGM optimizes the baseline model in humor detection, humor rating, and offensive rating tasks. and based on FGM, Bi-LSTM does more better in these tasks. Because Bi-LSTM can extract sentence features in more detail, especially bidirectional sequence features. Experiments show that these features are more conducive to downstream tasks.

Finally, Compare Method only optimizes the offensive rating predict task, but it not good at humor rating predict task, we think the non-humorous sentences. We speculate that non-humorous sentences with a 0 rating interferes with the comparison of two randomly selected sentences in compare method. The number of sentences that select non-humorous sentences for comparison is too large to help the model predict rating, so the auxiliary task interferes with the baseline model.

6 Conclusion

This paper introduces the experiment in SemEval-2021 task 7 HaHackathon: Detecting and Rating Humor and Offense. In this article, we propose two main assumptions. The first point is that the model is difficult to obtain the real meaning of the tag according to the change of the 0-5 rating. So the method of adding the auxiliary task on the baseline model was proposed. The auxiliary task is comparing different sentences according to the number proposed by us all to strengthen and supplement this process. This method does the best in offensive rating predict task, achieve 0.4761 RMSE. Second, the output of the pre-training model is similar to the word vector, which needs further processing to be more suitable for downstream tasks. So we try to use Bi-LSTM. Indeed Bi-LSTM does the best, achieve the 95.05% F1-Score in the humor detection task, and 0.6143 RMSE in the humor rating task. These approaches do not play an optimized role in the controversy detection task. The baseline does the best, achieve the 61.74% F1-score. The main reason for this problem lies in the interference

Model	Humor F1	Humor RMSE	Controversy F1	Offensive RMSE
Final results	0.9386	0.6312	0.5455	0.4761 RoBERTa
0.9344	0.6961	0.6174	0.5146	
FGM	0.9481	0.6311	0.5614	0.4847
Bi-LSTM+FGM	0.9505	0.6143	0.5609	0.4956
FGM+FocalLoss	0.9386	-	0.5454	-
Compare Method	-	0.6906	-	0.4761

Table 1: The result of several optimize approach on test datasets

of non-humorous sentences. So there is still room for improvement, such as eliminating the influence of non-humorous sentences, adjust the model parameters and try other pre-trained models. Or try to use a classification model and regression model in machine learning, such as Bayesian or CRF, to process the output of BERT. Therefore, the future work is to find a better way to remove the influence of non-humorous sentences and find a better way to optimize the controversy detection task. And then do more experiments to get better results.

References

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*.

Lily Li, Or Levi, Pedram Hosseini, and David A. Broniatowski. 2020. A multi-modal method for satire detection using textual and visual cues.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

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*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26:3111–3119.

Takeru Miyato, Andrew M. Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. In *International Conference on Learning Representations*.

J. X. Morris, E. Lifland, J. Y. Yoo, J. Grigsby, D. Jin, and Y. Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

F Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. Mcdaniel. 2017. Ensemble adversarial training: attacks and defenses.

Jin Wang, Bo Peng, and Xuejie Zhang. 2018. Using a stacked residual LSTM model for sentiment intensity prediction. *Neurocomputing*, 322:93–101.

Orion Weller, Nancy Fulda, and Kevin Seppi. 2020. Can humor prediction datasets be used for humor generation? humorous headline generation via style transfer. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 186–191, Online. Association for Computational Linguistics.