YoungSheldon at SemEval-2021 Task 7: Fine-tuning Is All You Need

Mayukh Sharma, Ilanthenral Kandasamy, W.B. Vasantha
School of Computer Science and Engineering
Vellore Institute of Technology
Vellore, Tamil Nadu, India
04mayukh@gmail.com, ilanthenral.k@vit.ac.in,
vasantha.wb@vit.ac.in

Abstract

In this paper, we describe our system used for SemEval 2021 Task 7: HaHackathon: Detecting and Rating Humor and Offense. We used a simple fine-tuning approach using different Pre-trained Language Models (PLMs) to evaluate their performance for humor and offense detection. For regression tasks, we averaged the scores of different models leading to better performance than the original models. We participated in all SubTasks. Our best performing system was ranked 4 in SubTask 1-b, 8 in SubTask 1-c, 12 in SubTask 2, and performed well in SubTask 1-a. We further show comprehensive results using different pre-trained language models which will help as baselines for future work.

1 Introduction

Humor is an intelligent form of communication with the capability of providing amusement and provoking laughter (Chen and Soo, 2018). It helps in bridging the gap between various languages, cultures, and demographics. Humor is a very subjective phenomenon. It can have different intensities, and people may find some jokes funnier than others. In certain situations, some jokes may be offensive to a certain group of people. All these characteristics of humor pose an interesting linguistic challenge to NLP systems. SemEval 2021 Task 7: HaHackathon: Detecting and Rating Humor and Offense (Meaney et al., 2021) aims to draw attention to these challenges in humor detection. The task provides a dataset of humorous content annotated using people representing different age groups, gender, political stances, and income levels. The content of the provided dataset was in English.

Participating in all SubTasks, we propose a fine-tuning based approach on pre-trained language models. Pre-trained Language Models learn syntactic and semantic representations by training on large amounts of unsupervised data. Recently there has been a lot of interest in PLMs. Researchers have come up with different pre-training methods using Auto Encoding(AE) and Auto-Regressive(AR) language modeling techniques. Often these pre-trained models contain millions of parameters and are computationally expensive. Fine-tuning different models may lead to different results on downstream tasks. This makes the choice of PLM an important factor. We present a comparative study of different PLM models and their performance in all SubTasks of SemEval 2021 Task 7: HaHackathon.

Our proposed fine-tuning approach for each PLM made use of a single layer of one neuron stacked on the PLM features. We performed experiments using BERT(Devlin et al., 2019), ELECTRA(Clark et al., 2020), RoBERTa(Liu et al., 2019), XLNet(Yang et al., 2019), MPNet(Song et al., 2020), and ALBERT(Lan et al., 2020). For regression tasks, we also used averaging technique, which we describe later in the paper. Our model performed well in SubTask 1-b and 1-c, achieving a rank of 4 and 8 respectively on the official leader board. For SubTask 2, our proposed averaging technique outperformed individual fine-tuned models by a good margin and was ranked 12. Our code is available online[1] for method replicability.

2 Background

The task of automatic humor recognition refers to deciding whether a given sentence expresses humor. The problem of humor recognition is often formulated as a binary classification problem aiming to identify if the given text is humorous. (Weller and Seppi, 2019) performed a study to identify if a joke is humorous or not using transformers

[1]https://github.com/04mayukh/YoungSheldon-at-SemEval-2021-Task-7-HaHackathon

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 1146–1152
Bangkok, Thailand (online), August 5–6, 2021. ©2021 Association for Computational Linguistics

(Vaswani et al., 2017). They used the body of the joke, the punchline exclusively, and both parts together. Combining both parts lead to better performance and it was found that a punchline carries more weight than the body of a joke for humor identification. (de Oliveira and Rodrigo, 2015) experimented with SVM's, RNN's, and CNN's for identifying humor in Yelp reviews using a bag of words and mean word vector representations. (Chen and Soo, 2018) used CNN-based models for identifying humor content. (Annamoradnejad, 2021) uses a neural network built on BERT embeddings learning features for sentences and whole text separately and then combining them for prediction on 200k Short Texts for Humor Detection dataset on Kaggle[2]. Binary classification tasks help us to separate humorous content but are unable to quantify the degree of humor. SemEval-2017 Task 6: #HashtagWars (Potash et al., 2017) aimed to study the relative humor content of funny tweets by either generating the correct pairwise comparisons of tweets (SubTask A) or finding the correct ranking of the tweets (SubTask B) based on their degree of humor content. SemEval-2020 Task 7: Assessing Humor in Edited News Headlines (Hossain et al., 2020) presented a study on editing news headlines to make them humorous. The task involved quantifying the humor of the edited headline on a scale of (0-3) as well as comparing the humor content of the original and edited headline. SemEval-2020 Task 8: Memotion Analysis- The Visuo-Lingual Metaphor! (Sharma et al., 2020) provides details on humor classification as well as predicting its semantic scale on internet memes using both images and text. OffensEval (Zampieri et al., 2020) (Zampieri et al., 2019) provides insights for identifying offensive content on social media.

Humor is an intelligent way of communication in our daily lives. It helps bridge the gap between people from various cultures, ages, gender, languages, and socioeconomic status making it a powerful tool to connect with the audience. Humor is a highly subjective phenomenon. People from different demographics may have a different perception of humor, and some may even find it offensive. This makes identifying humor a tough task. SemEval 2021 Task 7: Hahackathon: Linking humor and offense across different age groups aims to study this subjective nature of humor, which has two SubTasks which we describe as:

SubTask 1: Given a labeled dataset D of texts, the task aims to learn a function that can:

- SubTask 1-a: predict if a text is humorous or not.
- SubTask 1-b: quantify humor present in a humorous text within a range of (0-5).
- SubTask 1-c: predict if the humor rating would be controversial for a humorous text, i.e., the variance of the rating between annotators is higher than the median.

SubTask 2: Given a labeled dataset D of texts, the task aims to learn a regression function that can quantify how offensive a text is for general users within a range of (0-5).

Dataset Statistics: Table 1 represents the dataset statistics for classification tasks. For SubTask 1-a we can see there is a slight class imbalance between humorous and non-humorous labels. We overcome the problem of class imbalance using class weights which we define as: Let X be the vector containing counts of each class X_i where $i \in X$. Then the weights for each class were given as:

$$weight_i = \frac{max(X)}{X_i + max(X)}$$

For SubTask 1-c the label distribution was balanced. Table 2 represents the statistics for the regression tasks. Another observation on the training set for SubTask 2 was that 3388 samples had an offensive rating of 0 and nearly 80% of samples had offense rating in the range 0-1.

3 System Overview

3.1 Pre-trained Language Models

NLP being a diverse field contains many tasks, but most task-specific datasets contain only a few hundred or a thousand human-labeled samples. To overcome this problem, researchers have come up with a method called pre-training (Qiu et al., 2020) which involves training general-purpose language representation using enormous amounts of unannotated textual data. These language models can then be fine-tuned on various downstream tasks and have shown promising results in many natural processing tasks (Dai and Le, 2015; Peters et al., 2018; Radford and Narasimhan, 2018). Next, we briefly discuss some pre-trained language models we used for the task.

[2]https://www.kaggle.com/moradnejad/200k-short-texts-for-humor-detection

		Humorous	Non-humorous	Total
	Train	4932	3068	8000
Task 1-a	Validation	632	368	1000
	Test	615	385	1000

		Controversial	Non-controversial	Total
	Train	2465	2467	4932
Task 1-c	Validation	308	324	632
	Test	279	336	615

Table 1: Dataset statistics for classification tasks.

		Mean	Standard Deviation	Count
	Train	2.260	0.566	4932
Task 1-b	Validation	2.269	0.572	632
	Test	2.119	0.546	615

		Mean	Standard Deviation	Count
	Train	1.393	1.185	8000
Task 2	Validation	0.706	1.190	1000
	Test	0.480	0.830	1000

Table 2: Dataset statistics for regression tasks.

3.2 Brief overview of used Pre-trained Models

BERT: Bidirectional Encoder Representations from Transformers is a bi-directional language model that uses Transformer (Vaswani et al., 2017) architecture to learn contextual relations between different words in a text sequence (Devlin et al., 2019). It makes use of two training strategies i.e., Masked Language Modelling (MLM) and Next Sentence Prediction (NSP).

ELECTRA: It introduces a new pre-training objective called Replaced Token Detection (RTD) (Clark et al., 2020). Unlike BERT which introduces <MASK> tokens, Electra replaces certain tokens with plausible fakes. The pre-training task then requires the model to determine if the input tokens are the same or have been replaced. This binary classification task is applied to all tokens unlike the small number of masked tokens making RTD more efficient than MLM.

RoBERTa: A Robustly Optimized BERT Pretraining Approach (Liu et al., 2019) was developed by Facebook. They made use of the BERT architecture with modifications to improve the performance on downstream tasks. They made use of dynamic masking in the pre-training objective and removed the NSP objective. They also trained the model for a longer duration with more data and a larger batch size. They outperformed BERT on several downstream tasks.

XLNet: XLNet (Yang et al., 2019) is a generalized autoregressive pre-training method that takes the best of both AR language modeling and AE modeling techniques. It proposed a permutation language modeling objective for pre-training that helps learn bidirectional contexts. It also helps overcome the pretrain-finetune (Yang et al., 2019) discrepancy present in BERT due to its autoregressive formulation.

MPNet: MPNet (Song et al., 2020) was proposed by Microsoft. It overcomes the positional discrepancy between pre-training and fine-tuning in XLNet which does not use the full position information of a sentence. It proposes a unified view of masked language modeling and permuted language modeling by rearranging and splitting the tokens into predicted and non-predicted parts. It uses MLM and PLM to model the dependency among predicted tokens and see the position information of the full sentence.

ALBERT: A Lite BERT for self-supervised learning of language representations (Lan et al., 2020) is a modification of BERT aiming to efficiently allocate the model capacity to help reduce training time and reduce memory consumption. ALBERT decomposes the embedding matrix into a lower dimension which is then projected to the hidden space. This is called factorized embedding parameterization and helps in reducing the parameters. It also makes use of layer sharing across all layers which helps remove redundancy. Additionally, it uses inter-sentence coherence loss based on Sentence Order Prediction (SOP) (Lan et al., 2020).

3.3 Fine-tuning

We fine-tuned the pre-trained language models for each SubTask by stacking a dropout layer followed by a single neuron dense layer on top of PLM features. We used the features of [CLS] token in the case of ALBERT, BERT, XLNet, ELECTRA, and start token (<s>) features in the case of RoBERTa, and MPNet. For the classification task, sigmoid activation was used in the final layer. For the regression task, we did not use any activation. Negative values were converted to zero in regression tasks.

3.4 Averaging for Regression tasks

For regression tasks, we combined all fine-tuned models by averaging their predictions. For SubTask 1-b, we averaged the predictions of all models. For SubTask 2 as stated earlier, there were many zero values in the training set therefore, we averaged the predictions only when all models predicted a non-zero value. If any of the models predicted zero for a given sample, we took zero as the final prediction.

4 Experimental Setup

We used ekphrasis (Baziotis et al., 2017) library for pre-processing the text inputs. It normalized date, time, numbers to a standard format and also performed spelling correction. For tokenization, we used Hugging Face's (Wolf et al., 2020) implementation of fast tokenizers for each pre-trained model. We fixed the sequence length of samples to 150 tokens. Models were developed on Keras[3] (Chollet et al., 2015) using the transformers[4] (Wolf et al., 2020) library by Hugging Face. We used Adam (Kingma and Ba, 2017) optimizer for fine-tuning. Learning rate of 1e-4 was used for ELECTRA. For other models, we experimented with 1e-5, 2e-5, and 3e-5. We used binary cross-entropy loss for classification tasks and logCosh loss for regression tasks. Batch size of 16 was used for all models. Fine-tuning was performed on TPU's on Google Colab. We fine-tuned for four epochs on SubTask 1 and 8 epochs on SubTask 2. F1 score and RMSE were used as an evaluation metric for classification and regression tasks. Weights with the best performance on the development set were used for making predictions on the test set.

[3]https://keras.io

[4]https://huggingface.co/transformers

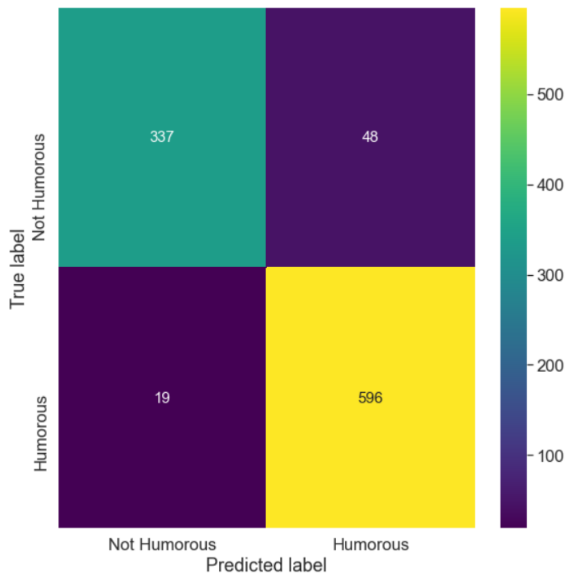

Figure 1: Confusion matrix for SubTask 1-a.

5 Results and analysis

Table 3 shows the results of our proposed fine-tuning approach for different pre-trained models. Our simple averaging technique worked quite well for regression tasks. Our model was ranked 4 in SubTask 1-b and ranked 12 in SubTask 2. The averaging method proposed by us for SubTask 2 provided a significant improvement in the RMSE score against the individually pre-trained models. Upon examination of the test set, we found 40.8% of samples were given zero offense rating. Thus, our decision to predict zero if any of the model predicted zero helped in improving scores against individual models. For classification tasks, our model was ranked 8 in SubTask 1-c and performed well for SubTask 1-a.

Figure 1 and Figure 2 show plots of confusion matrices for our best performing model fine-tuned on BERT. For Subtask 1-a, our model was efficient in separating the humorous and non-humorous content as false positives are low for each class. For SubTask 1-c, our model performed well in identifying the controversial text but did not perform

Model	Precision		Recall		F1		Accuracy	
Subtask	1-a	1-c	1-a	1-c	1-a	1-c	1-a	1-c
BERT-base	.9254	.4630	**.9691**	**.9426**	**.9467**	**.6210**	**.9330**	.4780
ELECTRA-base	.9269	.4747	.9073	.7419	.9170	.5790	.8990	.5105
RoBERTa-base	.9518	.4708	.8991	.8960	.9247	.6172	.9100	.4959
MPNet-base	**.9658**	**.4879**	.9203	.7275	.9425	.5841	.9310	**.5300**
XLNet-base	.9489	.4812	.9365	.6881	.9427	.5663	.9300	.5219
ALBERT-large	.9632	.4666	.8943	.8530	.9274	.6032	.9140	.4910

Table 3: Test set results for SubTask 1-a and SubTask 1-c.

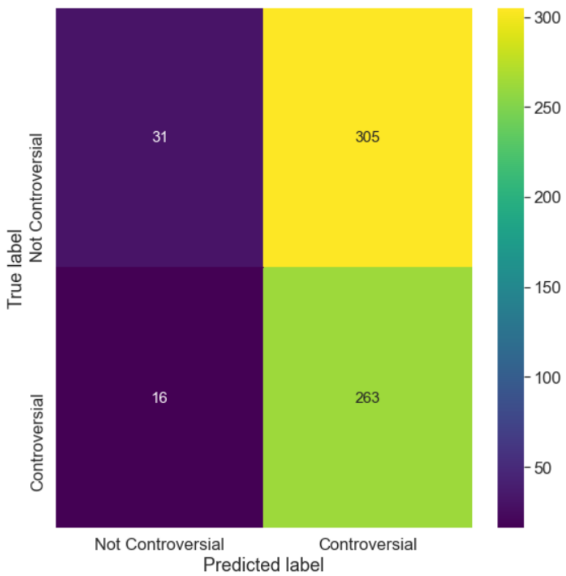

Figure 2: Confusion matrix for SubTask 1-c.

Model	RMSE	
SubTask	Task 1-b	Task 2
BERT-base	.5380	.5066
ELECTRA-base	.5418	.6071
RoBERTa-base	.5428	.5046
MPNet-base	.5401	.5142
XLNet-base	.5380	.5298
ALBERT-large	.5307	.5004
PLM Average	**.5257**	**.4499**

Table 4: Test set results for SubTask 1-b and SubTask 2.

very well for non-controversial text. The model has a very high recall but low precision due to high false positives for the controversial class which is evident from the confusion matrix. BERT, MPNet, and XLNet performed better than other PLMs for SubTask 1-a. For SubTask 1-b individual models had a similar performance. Averaging helped in improving the performance. BERT, ELECTRA, and ALBERT had the best performance on the test set for SubTask 1-c.

6 Conclusion

The paper describes our system used for competing in all SubTasks of SemEval 2021 Task 7: HaHackathon: Detecting and Rating Humor and Offense. We used a simple fine-tuning approach for analyzing the performance of various pre-trained language models for the task of humor detection. We performed well in all SubTasks except SubTask 1-a. A lot of research is happening around pre-trained language models with new and better models coming up. These models are large and computationally expensive. Choosing a model becomes a difficult task as they may have different results on downstream tasks. We, therefore, performed experiments with the recent state-of-the-art models and provide a comparative analysis of their performance. In the future, we would like to work on the effect of pre-training PLMs with additional task-specific data and then fine-tuning to see their performance on downstream tasks.

References

Issa Annamoradnejad. 2021. Colbert: Using bert sentence embedding for humor detection.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. 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)*, pages 747–754, Vancouver, Canada. Association for Computational Linguistics.

Peng-Yu Chen and Von-Wun Soo. 2018. Humor recognition using deep learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Pa-*

pers), pages 113–117, New Orleans, Louisiana. Association for Computational Linguistics.

François Chollet et al. 2015. Keras. `https://keras.io`.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. 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 the Fourteenth Workshop on Semantic Evaluation*, pages 746–758, Barcelona (online). International Committee for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

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*.

Luke de Oliveira and A. Rodrigo. 2015. Humor detection in yelp reviews.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2017. SemEval-2017 task 6: #HashtagWars: Learning a sense of humor. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 49–57, Vancouver, Canada. Association for Computational Linguistics.

XiPeng Qiu, TianXiang Sun, YiGe Xu, YunFan Shao, Ning Dai, and XuanJing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897.

A. Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Chhavi Sharma, Deepesh Bhageria, William Scott, Srinivas PYKL, Amitava Das, Tanmoy Chakraborty, Viswanath Pulabaigari, and Björn Gambäck. 2020. SemEval-2020 task 8: Memotion analysis- the visuo-lingual metaphor! In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 759–773, Barcelona (online). International Committee for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *NeurIPS 2020*. ACM.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Orion Weller and Kevin Seppi. 2019. Humor detection: A transformer gets the last laugh. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3621–3625, Hong Kong, China. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Marcos Zampieri, Preslav Nakov, Sara Rosenthal, Pepa Atanasova, Georgi Karadzhov, Hamdy Mubarak, Leon Derczynski, Zeses Pitenis, and Çağrı Çöltekin. 2020. SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1425–1447, Barcelona (online). International Committee for Computational Linguistics.