SemEval-2020 Task 7: HaHackathon: Detecting and Rating Humor and Offense

Aakash Khandelwal **Nitik Jain** **Rishabh Sanjay** **Vijit Malik**

Indian Institute of Technology, Kanpur

Abstract

Automatic humor/offence detection has interesting use cases in modern technologies, such as chatbots and personal assistants. The task in SemEval-2021 workshop challenges the participants to automate the process of detecting the extent humour and offence in textual conversations. In this report, we evaluate various approach for detecting humor and offence in short texts. We build up from evaluating classical models for classification tasks to neural and transformer based models, carefully analyzing the feasibility and performance of each. The best F1 score we obtained in the task of humor detection is of 0.932, and the best RMSE we obtained for rating humor is 0.446. We also talk a little about the explainability perspective of our models.

Disclaimer

All the work carried out in the project has not been re-used from any another course project at IITK or elsewhere. We have properly cited the portions of code and images that have been used from elsewhere in the project.

1 Introduction

The past decades have seen an explosive growth of user-generated content through social media platforms. Expressions of opinions and feelings through these platforms has become quite common. Tracking and analyzing public opinions from social media can help to predict certain political events or predicting people's attitude towards certain products. Therefore, detecting sentiments and emotions in text have gained a considerable amount of attention.

The shared task (Task 7: "HaHackathon") in SemEval-2021 workshop has been designed for detecting humor and offense in textual conversations. In this task, the participants are given a textual dialogue i.e. a user utterance along with three turns of context. The participant teams have to classify whether the utterance is humorous or not. Along with this, there is regression task of how humorous and how offensive the statement. Further details about the Task and the datasets are presented in Section 3. This report describes our team approach to detect humor and humor rating.

2 Related Work

Human Emotions are basic traits and have been studied by researchers in the various domains of science for several years. Prominent works like Ekman's six class categorization of emotions (Ekman, 1992) and Plutchik's "Wheel of Emotion" (Plutchik and Kellerman, 1986) which suggested eight primary bipolar emotions are considered the base of current research. A number of machine learning approaches have been used to detect and predict emotions and sentiments. With the rise of deep learning methods which strongly rely upon the capabilities of Graphics Processing Units (GPU). ((Abdullah et al., 2018) (Dos Santos and Gatti, 2014)). So we experimented with Transformer models like BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019) and XLNet (Yang et al., 2019) as these have produced state of the art results in almost every task in recent times. The problem of detecting emotions is not straightforward in absence of facial expressions and voice modulations. With the challenges of NLP domain in general, the context of the conversation plays an important role in determining the actual emotion of the statement.

3 Task Description

This is a SemEval-2021 task. It consists of 2 major tasks:

Task 1 consists of humour detection task in which all the ratings were averaged to provide mean classification and rating scores. This tasks has 3 sub tasks under it:

- **Subtask 1a**: This is a binary classification task. We need to predict if a particular text is considered humorous(for an average user).

- **Subtask 1b**: This is a regression task. In this task we need to predict how humorous the text is, if it was classified as humorous by previous sub task. The values vary from 0-5.

- **Subtask 1c**: In this task we need to predict if the humour rating will be considered controversial i.e the variance of the rating between annotators is higher than the median. This task also will be carried out only on those text which were classified as humour by 1^{st} subtask. This is also a binary classification task.

Task 2 aims to predict how offensive a text would be (for an average user) with values ranging between 0-5.

- **Subtask 2a**: This task is mainly to predict how generally offensive a text is for users. This score is calculated regardless of whether the text was classified as humorous or not.

For our CS771 project we would be mainly focusing on subtask 1a and subtask 1b.

4 Corpus Description

The final dataset provided by the organizers contains one row for each text. 20 annotators were asked to annotate the data and give their ratings and votes. The dataset contains 8000 distinct rows (examples), where each row has a unique identifier, the text and the following values which needs to be predicted:

- **is_humour:** Binary in nature, based on majority class given by 20 annotators

- **humor_rating:** Ranging between 0-5 (both inclusive), basis the average rating

- **humor_controversy:** Binary, '1' if the variance of the rating by annotators is higher than the median variance, '0' otherwise

- **offense_rating:** Ranging between 0-5 (both inclusive), average of the offence rating

5 Models

5.1 Baseline Models

For baseline models, **nltk** library was used for tokenization Following classic Machine Learning Algorithms were applied for subtask 1a and subtask 1b using the **scikit-learn** library after extracting GloVe embeddings.

- **Subtask 1a**: Logistic Regression, Support Vector Machines(SVM), k-Nearest Neighbors, Decision Trees, Random Forests, XG Boost and AdaBoost.

- **Subtask 1b**: Linear Regression and MLP.

5.2 RNN based Models

Logits were extracted for the following models for subtask 1a. Basic Regression models were applied on the logits were applied for subtask 1b.

- **2-layered BiGRU model**: 2-level BiGRU layer was applied on the input sentence and the context vector was then used to compute the logits after passing through fully connected and softmax layer.

- **2-layered BiGRU model with Attention**: 2-level BiGRU layer was applied on the input sentence along with the Attention Mechanism with an intention to focus on the parts of the sentence which are responsible for generating humor. Attention weights so found were then again passed into a linear and a softmax layer to generate logits.

5.3 CNN based Models

We used a sequence of 4 convolutional 1d layers + max pooling layer and in the end followed by a linear layer. This model was applied on the input sentence and then passed through the softmax layer to generate the logits.

5.4 Transformer Based Models

Attention is all you need (Vaswani et al., 2017) gave rise to numerous Transformer based models which eventually led to the new era of NLP. We will be using the following Transformer based models and as in the previous section will apply regression on the logits for subtask 1b.

- **BERT, RoBERTa, XLNet**: Output of the token corresponding to CLS label, from the final layer in each of the models was used for

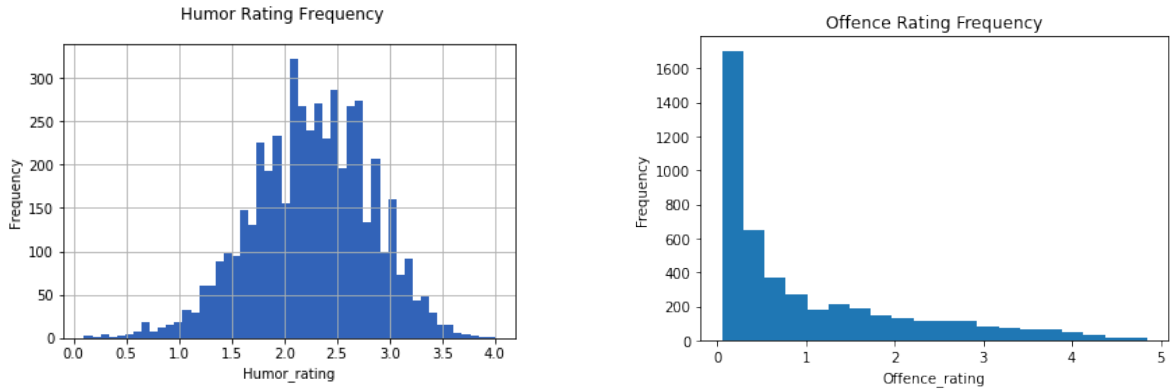

Figure 1: Comparison of humor and Offense ratings across the train set.

	text	is_humor	humor_rating	humor_controversy	offense_rating
95	Sweet potatoes r rich in beta-carotene helps balance skins pH to combat dryness promotes cell turnover health skin potatoes	0			0
96	Just found out my wife has cooties Im headed to the clinic to get tested So many emotions right now	1	2	1	0
97	How many gay men does it take to screw in a light bulb Only one but it takes the entire emergency room to remove it	1	2.74	1	2.95
98	What do the Toronto Maple Leafs and the Titanic have in common Both look good until they hit the ice	1	2.3	0	0.3
99	A family of Jews sit on the sofa at home When its cold they sit around a candle When its really really cold they light it	1	2.2	0	2.4

Figure 2: Example inputs from given dataset.

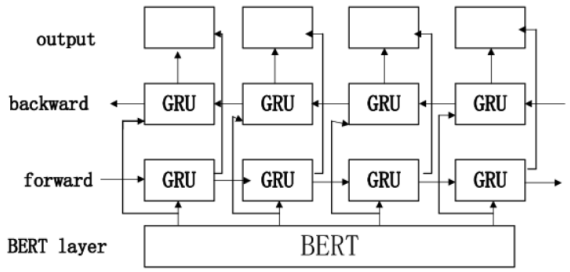

Figure 3: Architechture of BERT with BiGRU on top. This image was taken from (Huang et al., 2019)

obtaining logits after passing through a fully connected and a softmax layer.

5.5 Ensemble Models

With our top 3 best performing models on the subtask 1a, namely BERT, XLNet and RoBERTa, we created a voting ensemble to attain a boost in our accuracy.

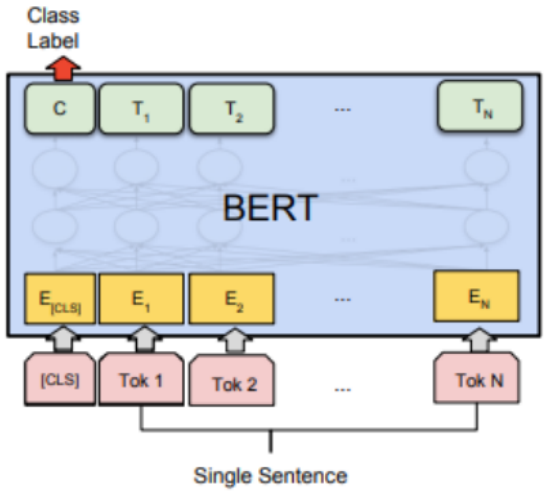

Figure 4: BERT for sequence classification architecture[1]

[1]This image was taken from http://jalammar.github.io/illustrated-bert/

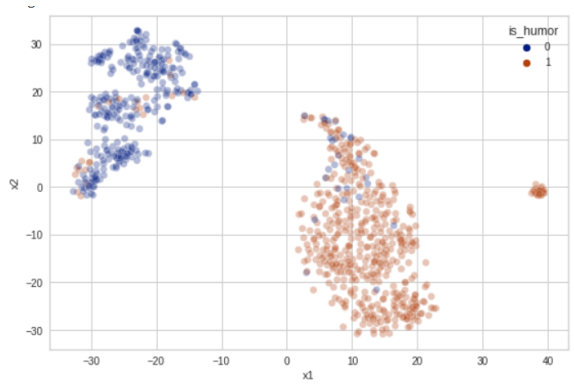

Figure 5: BERT features' 2-dimensional tSNE visualization.

6 Experiments

We run a battery of experiments on our data. We have used the train data of the competition and split it into train, development and test set. We run various classical Machine learning models for the humor detection task(Subtask 1(a)). For results of classical models refer to table 1.

We experimented with various neural models like MLP, CNNs, Recurrent Neural Networks and Transformer based models. We used Pytorch to implement our MLP, CNNs and RNN based neural models. Transformer based models have provided State of the art results on almost all NLP tasks recently. We used the implementations provided by the HuggingFace[2] library for BERT for sequence classification and similarly for XLNet and RoBERTa. For the architecture refer to figure 4. We used the embeddings corresponding to the CLS tokens and then added a linear layer with a dropout and softmax over them. The CLS token is a special token of the Transformer model which is basically used for classification tasks. For the BERT+BiGRU model, we used the BERT embeddings corresponding to all the tokens of an input sentence and then passed BiGRU over these sequence of embeddings. For the entire architecture of BERT+BiGRU refer to figure 3. The results by these models are provided in table 2.

For the regression task(Subtask 1b) we used the BERT model fine tuned over subtask 1(a) to get the CLS embeddings for each sentence. These were used as feature representation of whole sentence that also capture the humorous properties of the text. We used these vectors and then ran Ridge Regression and MLP over them. Another MLP model was trained over the PCA features of the BERT features. The results of these are provided in table 3.

7 Results and Error Analysis

We used only the training data provided to us by the organizers and splitted it into train, test and development sets. Please note that all the results are provided on same test split.

7.1 Subtask 1(a)

Among the classical models, XGBoost with 150 estimators performs the best with an F1 score of 0.898 (see table 1). This is interesting to see as XGBoost is able to outperform the RNN based models like BiGRU which provided us with an F1 score of 0.858(see table 2). It is also interesting to note that on our dataset, training a BiGRU model with an added attention layer on top decreases the model performance.

As expected from transformer based models, they provided us with the best results. The BERT large model attains a higher F1 score of 0.929 when compared with it's XLNet and RoBERTa mutations. It is interesting to note that although RoBERTa is pre-trained on 10 times more dataset that what BERT was trained on, and XLNet's advanced Permutation based language modelling, BERT was able to outperform them both in our task.

Training BERT with a BiGRU on top, ended up with the best f1 of 0.932 among all our models on Subtask 1(a).

We also visualized the features extracted from the fine-tuned BERT large model. We used these features and applied tSNE technique to visualize them in 2-dimensional space (see fig. 5). Note that BERT is able to separate out the two classes of humor and not humor quite significantly. Also, observe that there is a separate cluster of points of humorous class (class 1). We are currently looking into this observation manually about what this separate cluster of data points represents.

To look inside the black box of these huge transformer models about what they are learning and what they consider important while performing a prediction, we used the Captum library[4]. Using the IG method for getting the explanations, we noticed some interesting parts of text upon which the BERT model was largely focusing upon. See fig. 6 some for such examples. We can see that the model is

[2] https://huggingface.co/

[4] https://captum.ai/

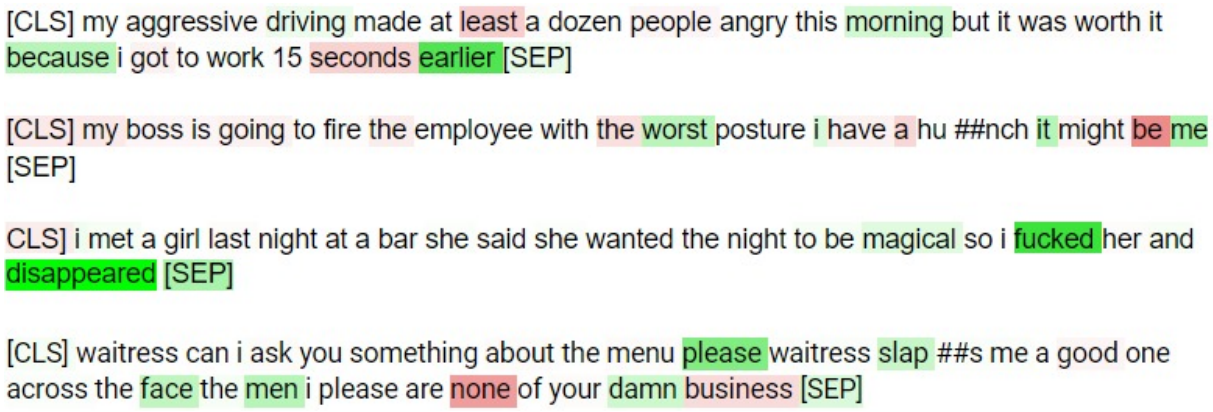

Figure 6: Explanations using Integrated Gradients method. The green colored tokens were relevant for making the prediction whereas the red colored tokens were considered irrelevant by the model. As the intensity of the color green increases, so does it's relevance score.

Model	Hyperparameters	Precision	Recall	F1	Acc. (%)
LR	solver='saga', L1_ratio=0.7	0.873	0.868	0.871	84.41
Decision Tree	default_params	0.781	0.751	0.765	71.66
Random Forest	max_depth=20	0.854	0.912	0.882	85.01
SVM	kernel= 'rbf'	0.900	0.880	0.890	86.64
XGBoost	n_estimators=150	**0.893**	**0.902**	**0.898**	**87.39**
AdaBoost	n_estimators=200	0.883	0.860	0.871	84.39
KNN	n_neighbors=28	0.926	0.791	0.853	83.27

Table 1: Results of Classical Models on Subtask 1(a)

Model	Hyperparameters[3]	Precision	Recall	F1	Acc. (%)
ANN (MLP)	lr = 5e-2, solver='lbfgs',max_itr=10000	0.914	0.882	0.898	87.62
CNN	lr = 5e-4, bs = 32, e = 6, k = 2, s = 1	0.869	0.851	0.859	86.89
BiGRU	lr = 5e-4, bs = 32, e = 5, h_dim= 64	0.881	0.869	0.875	88.26
BiGRU + Att.	lr = 5e-4, bs = 32, e = 5, h_dim = 64	0.853	0.863	0.858	86.27
BERT-large	lr = 2e-5, bs = 6, e = 5, seq_len = 128	0.926	0.932	0.929	93.26
RoBERTa-large	lr = 1e-6, bs = 6, e = 6, seq_len = 128	0.912	0.890	0.901	89.91
XLNet-large	lr = 2e-6, bs = 6, e = 6, seq_len = 128	0.905	0.927	0.916	92.01
BERT + BiGRU	lr = 2e-5, bs = 6, e = 5, n_layers = 1	**0.928**	**0.936**	**0.932**	**93.89**

[3]lr = learning rate, bs = batch size, e = epochs, k = kernal size, s = stride, h_dim = hidden layer dimension

Table 2: Results of Neural Models on Subtask 1(a)

Model	Hyperparameters	RMSE
BERT features + Ridge Regression	$\alpha = 100$	0.655
BERT features + MLP	4 units hidden layer + ReLU	0.479
BERT features + PCA + MLP	6 eigenvectors, 2 units hidden layer + ReLU	**0.446**

Table 3: Results of Neural Models on Subtask 1(b)

correctly attending upon the tokens that are crucial for text to be humorous.

7.2 Subtask 1(b)

Among Ridge Regression and Multi-layer Perceptron on BERT features with and without PCA, the least RMSE (Root Mean Squared Error) was observed in case of MLP with PCA, with a value of 0.446 (see table 3). We noted that the regression outputs from MLP of our test data had values from 1.27 to 2.35. This poses the problem of very low variance in our regression outputs, that is, the model predicts the values that are around the average of the range of true output values. This however was not observed in case of Ridge Regression.

8 Conclusion

After doing multiple experiments using various kinds of models, starting from classic Machine learning algorithms like SVM, KNN, etc, to Neural Networks, Recurrent Neural Networks like LSTM and we finally arrived at BERT-large model which gave the best result. We constructed 1024-dimensional BERT-embeddings from BERT-large model, which quite interestingly divides the dataset into two dense clusters using tSNE. We then developed a pipeline which predicts with 93.89% accuracy if the sentence is humorous or not. We then built a regression model to find a humor rating of the sentence. The best regression model that we could come up with was a MLP model with single hidden layer of 4 units with ReLU activation followed by the output layer. The input to this model for a sentence was the embedding of CLS token generated from our BERT model after passing the sentence through it. We achieved a best RMSE of 0.446 using a MLP model of 2 hidden units on PCA features of the BERT model output.

9 Future Work

As mentioned in Section 6.2 we will be looking into the separate cluster of data points that we have observed in fig. 5. We are planning to ensemble a variety of different types of models that we have trained so far to obtain a boost in F1 score.

Since the data given to us is very less, we are looking into augmenting the dataset either by using traditional NLP data augmentation strategies or by fine-tuning transformer models on other humor detection datasets and then on our dataset. We are also looking into using Twitter's sentiment based embeddings and PAD (Pleasure Arousal Dominance) values as additional features with BERT features for subtask 1(a).

As for Subtask 1(c), we were getting F1 score of around 0.5 when we trained a BERT large model on the classification of humor controversy. We assumed that there was some problem in our script but we recently found that all the techniques, even by the other teams are performing in a similar manner.

For Subtask 2, we will be first training a classification model for offense detection. In order to obtain true labels for this task, we will be taking a threshold above which the offense rating will be used to determine if the text is offensive or not. The rest will be the same as in Subtask 1(b).

10 Software Used and Code

10.1 Google Colab

Google Colab allows us to write and execute arbitrary code through our browser and is very well suited for Machine Learning and Deep Learning. The Colab notebooks are executed on Google's cloud, meaning we can use the power of Google hardware including GPU's and TPU's.

10.2 Scikit Learn

Scikit-learn is a free software machine learning library in Python programming language. It contains efficient implementations of various classification, regression and clustering algorithms such as LR, KNN, Random Forest, SVM, XGBoost etc.

10.3 Pytorch

PyTorch is an open source machine learning library used primarily for applications such as computer vision and natural language processing. A number of deep learning softwares are built on Pytorch including Hugging Face, Pytorch Lightning etc.

10.4 HuggingFace

HuggingFace is an open-sourced software which provides various NLP technologies. It is implemented using Pytorch. The HuggingFace Transformer library is an immensely popular Python library providing pretrained models that are useful in variety of NLP tasks.

10.5 Captum

Captum is a model interpretability library for PyTorch which currently offers a number of attribution algorithms that allows us to explain the predictions of deep learning models to some extent.

10.6 BERT CLS embedding extraction code

We used the code following link https://mccormickml.com/2019/05/14/BERT-word-embeddings-tutorial/ to extract the embedding corresponding to the CLS label for BERT for sequence classification model.

10.7 BERT fine-tuning code

We used the code from the following link https://mccormickml.com/2019/07/22/BERT-fine-tuning/ to fine tune our BERT for sequence classification model. Similarly XLNet and RoBERTa were also fine tuned.

References

Malak Abdullah, Mirsad Hadzikadicy, and Samira Shaikhz. 2018. Sedat: sentiment and emotion detection in arabic text using cnn-lstm deep learning. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 835–840. IEEE.

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*.

Cicero Dos Santos and Maira Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78.

Paul Ekman. 1992. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200.

H. Huang, X. Jing, Fei Wu, Yongfang Yao, Xinyu Zhang, and Xiwei Dong. 2019. Dcnn-bigru text classification model based on bert embedding. *2019 IEEE International Conferences on Ubiquitous Computing Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS)*, pages 632–637.

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. *arXiv preprint arXiv:1907.11692*.

Robert Plutchik and Henry Kellerman. 1986. Biological foundations of emotion. vol. 3 of emotion: Theory, research, and experience.

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.

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*, pages 5753–5763.