Humor@IITK at SemEval-2021 Task 7: Large Language Models for Quantifying Humor and Offensiveness

Aishwarya Gupta*, **Avik Pal***, **Bholeshwar Khurana***, **Lakshay Tyagi***, **Ashutosh Modi**
Indian Institute of Technology Kanpur (IIT Kanpur)
{aishwaryag20, avikpal, bholek, lakshayt}@iitk.ac.in
ashutoshm@cse.iitk.ac.in

Abstract

Humor and Offense are highly subjective due to multiple word senses, cultural knowledge, and pragmatic competence. Hence, accurately detecting humorous and offensive texts has several compelling use cases in Recommendation Systems and Personalized Content Moderation. However, due to the lack of an extensive labeled dataset, most prior works in this domain haven't explored large neural models for subjective humor understanding. This paper explores whether large neural models and their ensembles can capture the intricacies associated with humor/offense detection and rating. Our experiments on the SemEval-2021 Task 7: HaHackathon show that we can develop reasonable humor and offense detection systems with such models. Our models are ranked third in subtask 1b and consistently ranked around the top 33% of the leaderboard for the remaining subtasks.

1 Introduction

Like most figurative languages, humor/offense pose interesting linguistic challenges to Natural Language Processing due to its emphasis on multiple word senses, cultural knowledge, sarcasm, and pragmatic competence. A joke's perception is highly subjective, and age, gender, and socioeconomic status extensively influence it. Prior humor detection/rating challenges treated humor as an objective concept. SemEval 2021 Task 7 (Meaney et al., 2021) is the first humor detection challenge that incorporates the subjectivity associated with humor and offense across different demographic groups. Users from varied age groups and genders annotated the data with the text's humor and have provided an associated score for the same. It is also quite a generic phenomenon that a text might be humorous to one and normal/offensive to another. Rarely has it been noticed that the same content is globally accepted as witty. To the best of our knowledge, Meaney et al. (2021) is the first initiative towards annotating the underlying humor as controversial or not. Understanding whether a text is humorous and/or offensive will aid downstream tasks, such as personalized content moderation, recommendation systems, and flagging offensive content.

Large Language Models (LLMs) have recently emerged as the SOTA for various Natural Language Understanding Tasks (Lewis et al., 2019; Raffel et al., 2019; Conneau et al., 2019; Zhang et al., 2020). However, typical day-to-day texts, where these models have shown state of the art performance, are less ambiguous than texts having puns/jokes. Training and evaluating LLMs in the context of highly ambiguous/subjective English texts would serve as an excellent benchmark to figure out the current shortcomings of these models. This paper studies various large language models – BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), XLNet (Yang et al., 2019), ERNIE-2.0 (Sun et al., 2019) and DeBERTa (He et al., 2020) and their ensembles – for humor and offense detection tasks. Additionally, we explore a Multi-Task Learning framework to train on all the four sub-tasks jointly and observe that joint training improves the performance in regression tasks.

We have achieved significant performance on all the subtasks and have consistently ranked $\sim \frac{1}{3}^{rd}$ of the total submissions. We were ranked (1) 21^{st} with an F-score and accuracy of 94.8% and 95.81% respectively in Task 1a, (2) 3^{rd} with an RMSE score of 0.521 in Task 1b, (3) 9^{th} with an F-score and accuracy of 45.2% and 62.09% respectively in Task 1c; and (4) 16^{th} with an RMSE score of 0.4607 in Task 2. We release the code for models

* Authors contributed equally to the work. Names is alphabetical order.

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 290–296
Bangkok, Thailand (online), August 5–6, 2021.

and experiments via GitHub[1]

We organize the rest of the paper as: we begin with a description of the challenge tasks followed by a brief literature survey in section 2. We then describe all of our proposed models in section 3 with training details in section 4 and present the experimental results in section 5. Finally, we analyze our findings and conclude in section 6, and 7 respectively.

2 Background

2.1 Problem Description

SemEval 2021 Task 7: HaHackathon: Detecting and Rating Humor and Offense (Meaney et al., 2021) involves two main tasks – humor detection and offense detection. The organizers further subdivide the task into following subtasks:

1. Humor detection tasks:
   (a) **Task 1a** involves predicting whether a given text is humorous.
   (b) **Task 1b** requires predicting the humor rating of a given humorous text.
   (c) **Task 1c** incorporates humor subjectivity by posing a classification problem of predicting whether the underlying humor is controversial or not.
2. **Task 2** is an offense detection task and is posed as a bounded regression problem. Given a text, we need to predict a mean score denoting the text's offensiveness on a scale of 0 to 5, with 5 being the most offensive.

2.2 Related Works

Transfer Learning ULMFiT (Howard and Ruder, 2018) used a novel neural network based method for transfer learning and achieved SOTA results on a small dataset. Devlin et al. (2018) introduced BERT to learn latent representations in an unsupervised manner, which can then be finetuned on downstream tasks to achieve SOTA results. Lan et al. (2019); Liu et al. (2019); Sanh et al. (2019); Sun et al. (2019) have proposed several improvements to the BERT model. In this paper, we analyze the effects of using these different base models in the context of humor and offense detection.

[1] https://github.com/aishgupta/Quantifying-Humor-Offensiveness

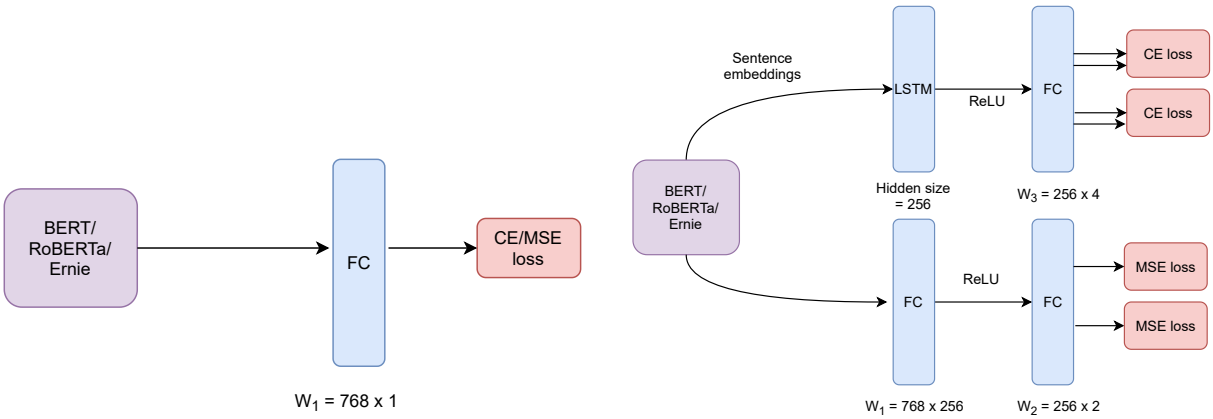
Humor & Emotion Detection Weller and Seppi (2019) first proposed the use of transformers (Vaswani et al., 2017) in humor detection and outperformed the state of the art models on multiple datasets. Ismailov (2019); Annamoradnejad (2020) extended the use of BERT models to humor classification. Fleșcan-Lovin-Arseni et al. (2017) did humor classification by comparing and ranking tweets while Docekal et al. (2020) edit the tweet and rank the extent of humor for the edited tweet on a scale of 0 to 3 (most funny). There has been extensive research in the area of text emotion prediction and generation (e.g., Witon et al. (2018); Colombo et al. (2019); Goswamy et al. (2020); Singh et al. (2021)). Demszky et al. (2020) curated a large scale emotion detection dataset and achieved SOTA results by finetuning a BERT model. However, none of these works delve into humor analysis' subjectivity, which is a prime focus of this task.

Sentiment and Pun Analysis Li et al. (2019); Maltoudoglou et al. (2020) study BERT based models for sentiment analysis. Ke et al. (2019) uses a combination of sentence embedding, POS tagging and word-level sentiment polarity scores for sentiment classification. Zhou et al. (2020) uses contextualized and pronunciation embeddings for each word and pass these through a neural network to detect and localize pun in the sentence. However, none of these works focus on the subjectivity of the underlying sentiment and pun in the text.

3 System Overview

3.1 Data

The challenge dataset comprises of a `train` set (labeled 8000 texts) and a `public-dev` set (labeled 1000 texts). Each text input is labeled as $1/0$ if it is humorous or not and rated with the offensiveness score on a scale of 0-5. If a text is classified as humorous, it is further annotated with humor rating and classified as controversial or not. For our single-task models (Section 3.2), we train on the `train` + `public-dev` set after obtaining a suitable stopping epoch by training and validating on the `train` and `public-dev` respectively. For our multi-task models (Section 3.3), we train on 8200 texts sampled randomly from `train` and `public-dev` sets and use remaining 800 text inputs for validation.



(a) Single-task model

(b) Multi-task model

Figure 1: Different Model architectures used for Humor/Offense detection/rating.

3.2 Single Task Model

As the tasks are evaluated independently, we have explored LLMs for each task/subtask independently and will be referring to them as single task models. Inspired by Demszky et al. (2020), for each task, we add a classification (for Task 1a, 1c) or a regression (for Task 1b, 2) head on top of the pretrained models like BERT, RoBERTa, ERNIE-2.0, DeBERTa and XLNet and train the model end-to-end (Figure 1a). This ensures that the model learns features solely related to the task, enhancing the performance. Also, as we only add a classification/regression head, the number of learnable parameters does not increase much. This helps us in finetuning the model on such a small dataset for a few number of epochs avoiding overfitting and resulting in better generalization.

3.3 Multi Task Learning

Collobert and Weston (2008) demonstrated that Multi-Task Learning (MTL) improves generalization performance across tasks in NLP. The different tasks though uncorrelated, share the same underlying data distribution. This can be of great help for tasks 1b and 1c where labeled instances are far less than for task 1a or 2. Exploiting the fact that all tasks share same data distribution, we propose to learn a model jointly on all the tasks. Specifically, we consider hard parameter sharing among differnet tasks and parameterize the base models using a neural network, followed by two heads for classification and regression tasks (Figure 1b). Our base model includes LLMs like BERT, RoBERTa, and ERNIE. Contrary to the LSTM layer, which helps in learning features using all the token level embeddings, the Fully Connected (FC) layer focuses only on the embedding of [CLS] token. Hence, having these two branches allow the model to focus on different tasks using the same sentence embedding and helps in learning enhanced embeddings for task 1b and 1c with much lesser labeled dataset.

3.4 Ensembles

Mostly LLMs differ in their training procedure, and architecture. These big language model frameworks are trained on wide set of datasets for a variety of tasks. Though, they all have comparable performance, they may still capture different aspects of the input. We try to leverage such varied informative embeddings based predictions by combining multiple models trained with different basenet using following strategies:

Jointly trained Model Embeddings: All the big language frameworks have shown huge performance improvement on multiple tasks owing to their highly informative latent input embeddings. We propose to learn an ensemble leveraging diverse aspects of the input captured by varied LLMs by concatenating their latent embeddings and mapping them to low dimensional space for task prediction. We use this method in learning ensembles of single task models explained in 3.2.

Aggregation of Trained Model Predictions: Joint-training though more informative and powerful, is a computationally intensive approach. Thus as an alternative, we use a weighted averaging of multiple pretrained models without compromising much on the performance.

1. **Weighted Aggregate of Regression Outputs:** For an ensemble of k models trained

Model	Task1-a		Task1-b	Task1-c		Task2
	F-Score	Accuracy	RMSE	F-Score	Accuracy	RMSE
STM (BERT)	-	-	0.5841	0.5934	0.4829	0.4997
STM (RoBERTa)	0.9523	0.9410	0.5929	**0.6242**	0.4536	-
STM (ERNIE-2.0)	0.9541	0.9430	0.5546	0.4113	0.5252	0.4716
STM (XLNet)	-	-	0.5656	0.5892	0.5171	-
STM (DeBERTa)	0.9532	0.9420	0.5491	-	-	-
STM (Agg. Ensemble)	**0.9581**	**0.9480**	0.5480	0.4520	**0.6209**	0.4750
MTM (BERT)	0.9374	0.9210	0.5794	0.5080	0.5496	0.5049
MTM (RoBERTa)	0.9477	0.9350	0.5873	0.5479	0.5170	0.5141
MTM (ERNIE-2.0)	0.9530	0.9420	0.5541	0.5389	0.5187	0.4961
STM + MTM (Agg. Ensemble)	0.9520	0.9400	**0.5210**	0.5321	0.5252	**0.4520**

Table 1: Metrics on the test dataset for the major models on all the sub-tasks. MTM stands for Multi-Task Model, STM stands for Single Task Model, and Agg. Ensemble is Aggregation Based Ensembling without having to jointly train all the models together.

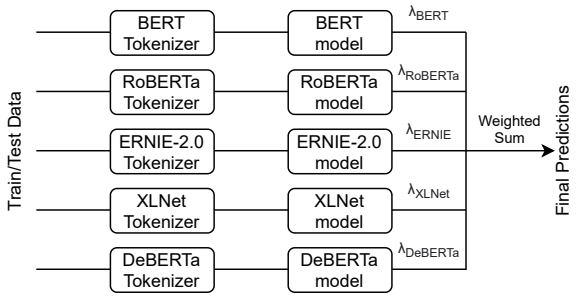


Figure 2: **Weighted-Average Ensembling:** The data is tokenized and then passed to the respective model. A weighted sum is done to obtain the final predictions. λ_i represents the weight for model i.

using different LLMs as basenet, the aggregate output $\hat{y}$ is computed as $\hat{y} = \sum_{i=1}^{k} \lambda_i \cdot \hat{y}_i$ where y_i and λ_i represents the output and weight of the i^{th} model respectively. The weights λ_i are obtained through extensive grid search on the held out validation dataset or set to a $\frac{1}{k}$ when trained on the entire dataset without a validation set. The complete approach is shown in figure 2.

2. **Voting Based Classification:** This is one of the most popular approach of learning an ensemble and does not involve any hyperparameters or retraining of any of the constituent models. This involves training multiple models independently and using maximum among all the predictions as the final output. For a binary classification task, the final output $\hat{y}$ is by max-voting across the independent models.

4 Experimental Setup

We used Pytorch (Paszke et al., 2019) and HuggingFace (Wolf et al., 2020) library for our models, and Google Colab GPUs for training and inference. We use ADAMW (Loshchilov and Hutter, 2019) and ADAM (Kingma and Ba, 2017) optimizer with initial learning rate of $2e^{-5}$ for training single task and multi task models respectively. For each of the models we follow a dedicated training pipeline described in subsequent sections.

4.1 Data preprocessing

We split the dataset into training and validation data as described in Section 3.1. The sentences are annotated with a [CLS] token in the beginning and given as an input to the model. We performed additional experiments by removing stopwords but noticed a slight deterioration in the performance.

4.2 Loss Functions

Task 1a & 1c are instances of binary classification problem and thus have been trained using cross-entropy loss. For predicting humor and offense rating i.e., Task 1b and 2, we have used mean squared error as the loss function.

4.3 Training Details

All the models are trained for n epochs where n is a hyper-parameter tuned on the validation set using early stopping criteria. For single task models, we split `train` data into training and validation set to learn the optimal value of n and then train the model from scratch on `train` + `public-dev`

Rank	Task1-a		Task1-b	Task1-c		Task2
	F-Score	Accuracy	RMSE	F-Score	Accuracy	RMSE
Rank-1	0.982	0.9854	0.4959	0.4943	0.6302	0.4120
Rank-2	0.975	0.9797	0.4977	0.4699	0.6279	0.4190
Rank-3	0.960	0.9676	0.5210	0.4699	0.6270	0.4230
Ours	0.948 (21)	0.9581 (21)	0.5210 (3)	0.452 (9)	0.6209 (9)	0.4607 (16)

Table 2: Comparison of our results with those on top of the leaderboard. (*) indicates our rank on the leaderboard in that task.

set for n epochs. In case of multi task models, all the tasks do not converge at the same rate. Thus, we train multi task models on randomly sampled 8200 texts from `train` + `public-dev` dataset and validate on the remaining 800 texts. We use early stopping criteria on validation dataset independently for each task.

5 Results

We have trained multiple single task and multi task models using basenet LLMs like BERT, DistilBERT, RoBERTa, XLNet, Albert (Lan et al., 2019), Electra (Clark et al., 2020), DeBERTa, and ERNIE-2.0. We also learned ensembles of single task models by either training a classification/regression head on concatenated input embeddings or using weighted aggregate of the models' predictions. Apart from this, we also explored voting based ensemble of multi-task models. All our models perform comparably on all tasks and the major models are reported in Table 1. We also compare our best model performance with the top 3 submissions on the leaderboard and report it in Table 2.

6 Analysis

6.1 Data Augmentation

One recurring issue across all our trained models is the high susceptibility to overfitting. Data Augmentation is a widely accepted solution to reduce overfitting by generating slight variants of the given dataset and is extremely useful for a smaller dataset.

One such approach is Masked Language Modelling (MLM), used to perform context-specific data augmentation (Ma, 2019) and has been used in training LLMs. However, following this data augmentation during training has consistently degraded the performance of our models. We hypothesize that this is due to the mismatch between the contextual meaning and the associated humor/offense. MLM-based augmentation strategies, with models pre-trained to preserve the sentence's meaning, fail to capture the associated humor/offense.

Often the selection of words in a sentence is responsible for its humor/offensive rating. Replacing such words by their synonyms can change the humor/offense rating substantially. Hence, using such a data augmentation approach during training will inject heavy noise in the ground truth resulting in deteriorated performance.

6.2 Correlation across Tasks

Contrary to our belief, we fail to ascertain any direct relationship between the humor controversy and the offense rating prediction task. We compute the mean offense rating for the texts labeled as controversial and for texts marked as non-controversial. The computed mean values are too close to each other to demonstrate any direct correlation conclusively.

6.3 Dataset Size

In literature, finetuning LLMs on small size task specific dataset has shown remarkable task performance. However, our single dedicated task models could not perform better than our multi-task model for Task 1b. We attribute this to relatively small size of supervised dataset available for Task 1b in comparison to other tasks. In our multi task models, though we have lesser labeled text for Task 1b, our sentence embeddings are still updated using the complete available dataset. Thus, our multi task model learns underlying distribution better than single task model owing to join learning and shared parameters for task 1b and 2. We believe that this is the main reason for the enhanced performance of our model on Task 1b which has lesser supervised data available in comparison to Task 1a or 2.

7 Conclusion

We have presented several experiments using large language models like BERT, XLNet, etc., and their ensembles for humor and offense detection and rating. We also discuss some of the underlying challenges due to the subjective nature of humor and offense detection task. Using these, we explain why standard training practices used to prevent overfitting, like data augmentation, do not work in this context. Our experiments suggest that even though these models can reasonably capture humor and offense, they are still far from understanding every intricacy arising out of subjectivity. To tackle some of the problems highlighted in this paper, a compelling direction would be online data augmentation by alternating between training the embeddings and generating new texts to preserve the humor/offensiveness. Additionally, pretraining these models on datasets annotated by diverse annotators to capture a more comprehensive world knowledge should further help in generalization.

References

Issa Annamoradnejad. 2020. ColBERT: Using BERT sentence embedding for humor detection.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 160–167, New York, NY, USA. Association for Computing Machinery.

Pierre Colombo, Wojciech Witon, Ashutosh Modi, James Kennedy, and Mubbasir Kapadia. 2019. Affect-driven dialog generation. 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 3734–3743, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A Dataset of Fine-Grained Emotions. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

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*.

Martin Docekal, Martin Fajcik, Josef Jon, and Pavel Smrz. 2020. Jokemeter at semeval-2020 task 7: Convolutional humor. *arXiv preprint arXiv:2008.11053*.

Iuliana Alexandra Fleșcan-Lovin-Arseni, Ramona Andreea Turcu, Cristina Sîrbu, Larisa Alexa, Sandra Maria Amarandei, Nichita Herciu, Constantin Scutaru, Diana Trandabăț, and Adrian Iftene. 2017. #WarTeam at SemEval-2017 task 6: Using neural networks for discovering humorous tweets. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 407–410, Vancouver, Canada. Association for Computational Linguistics.

Tushar Goswamy, Ishika Singh, Ahsan Barkati, and Ashutosh Modi. 2020. Adapting a language model for controlled affective text generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2787–2801, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced BERT with disentangled attention. ArXiv prepint.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Adilzhan Ismailov. 2019. Humor analysis based on human annotation challenge at iberlef 2019: First-place solution. In *IberLEF@ SEPLN*, pages 160–164.

Pei Ke, Haozhe Ji, Siyang Liu, Xiaoyan Zhu, and Minlie Huang. 2019. Sentilr: Linguistic knowledge enhanced language representation for sentiment analysis. *arXiv preprint arXiv:1911.02493*.

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. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019.

Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. 2019. Exploiting BERT for end-to-end aspect-based sentiment analysis. *arXiv preprint arXiv:1910.00883*.

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*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Lysimachos Maltoudoglou, Andreas Paisios, and Harris Papadopoulos. 2020. BERT-based conformal predictor for sentiment analysis. volume 128 of *Proceedings of Machine Learning Research*, pages 269–284. PMLR.

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*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Aaditya Singh, Shreeshail Hingane, Saim Wani, and Ashutosh Modi. 2021. An end-to-end network for emotion-cause pair extraction. *arXiv preprint arXiv:2103.01544*.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2019. Ernie 2.0: A continual pre-training framework for language understanding.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Orion Weller and Kevin Seppi. 2019. Humor detection: A transformer gets the last laugh. *"Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing"*.

Wojciech Witon, Pierre Colombo, Ashutosh Modi, and Mubbasir Kapadia. 2018. Disney at IEST 2018: Predicting emotions using an ensemble. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 248–253, Brussels, Belgium. 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. Huggingface's transformers: State-of-the-art natural language processing.

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.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Yichao Zhou, Jyun-Yu Jiang, Jieyu Zhao, Kai-Wei Chang, and Wei Wang. 2020. " the boating store had its best sail ever": Pronunciation-attentive contextualized pun recognition. *arXiv preprint arXiv:2004.14457*.