Amherst685 at SemEval-2021 Task 7: Joint Modeling of Classification and Regression for Humor and Offense

Brian Zylich **Akshay Gugnani** **Gabriel Brookman** **Nicholas Samoray**
University of Massachusetts Amherst
{bzylich,agugnani,gbrookman,nsamoray}@umass.edu

Abstract

This paper describes our submission to the SemEval'21: Task 7- HaHackathon: Detecting and Rating Humor and Offense. In this challenge, we explore intermediate finetuning, backtranslation augmentation, multitask learning, and ensembling of different language models. Curiously, intermediate finetuning and backtranslation do not improve performance, while multitask learning and ensembling do improve performance. We explore why intermediate finetuning and backtranslation do not provide the same benefit as on other natural language processing tasks and offer insight into the errors that our model makes. Our best performing system ranks 7th on Task 1b with an RMSE of 0.5339.

1 Introduction

With the advancement in deep learning methods, NLP tasks like sentiment analysis and opinion mining have achieved high accuracy, however detection of some salient forms of figurative language such as humor remain difficult tasks.

Being able to infer humor and offense with a high accuracy can help improve and lead to better performance on downstream applications, such as content moderation, sentiment analysis, etc. This would be useful for various downstream applications, such as understanding tweets, reviews and feedback. However, humor detection is not trivial.

What makes identifying humor hard? Humor can consist of styles ranging from sarcastic to slapstick comedy, and it factors in both individual preferences and underlying cultures. Context, sounds, and vision or any combination of these can be key in building to a punchline (Cai, 2019). Humor appreciation is also highly subjective, as age, gender, and socio-economic status often impact the perception of a joke. Meaney (2020) identify 3 ways that classification of humor is difficult:

(1) Humor can differ between cultures,
(2) Humor can also differ within cultures, and
(3) Humor differs within the same person.

Our contributions are as follows: **(1)** we explore whether intermediate finetuning on other humor and offense datasets is helpful for this task, **(2)** we seek to identify if backtranslation augmentation is useful for humor detection, **(3)** we show that multitask learning is helpful when classifying and scoring both humor and offense, and **(4)** we find that ensembling different language models leads to improved results on some tasks. The code for our experiments is available at our Github repository[1].

2 Related Work

The challenge of humor detection has gained traction since 2017. Meaney (2020), explains in their proposal that prior work has explored humor detection as an objective task, averaging all annotations for a joke, to produce a single classification or rating. This treats humor as an objective concept, which is not the case. This motivated their challenge for SemEval'21 (Meaney et al., 2021) to explore these dimensions of humor and offense. We discuss some of the previous efforts to explore humor and offensiveness in the following paragraphs.

Badlani et al. (2019) explains how text in reviews is quite complex as they can be sarcastic, humorous, or hateful. An ordinary sentiment analysis would fail to perform well in such cases. They first extract features pertaining to sarcasm, humor, hate speech, and sentiment, and then use these features to inform sentiment classification. Their work is quite sensitive to catching negative sentiment, however, it does not do as well when sentiment changes halfway through the text. It also does not address the subjectivity of humor.

[1] https://github.com/bzylich/humor-by-demographic

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 1190–1195
Bangkok, Thailand (online), August 5–6, 2021.

ColBERT (Annamoradnejad, 2020) is among the first to use BERT (Devlin et al., 2018) for humor detection, reaching 98% classification accuracy and outperforming variants using recurrent neural networks and convolutional neural networks. Mao and Liu (2019) is another work that uses BERT to classify if a tweet is a joke or not and predict how humorous the tweet is. The work of Weller and Seppi (2019) explores extending humor detection capability by trying to assess whether or not a joke is humorous. They use transformers to identify humorous jokes based on ratings from Reddit pages, reaching human-level performance.

Earlier work, like (Donahue et al., 2017), use recurrent deep learning methods with dense embeddings to predict humorous tweets. In order to factor both meaning and sound in their analysis, they use GloVe embeddings combined with a novel phonetic representation as input to an LSTM.

Hossain et al. (2020) hosted the SemEval'20 event for humor detection in news headlines. The event challenged participants to classify whether an original headline or an altered headline is funnier and rate the funniness of the edited headline on a 0-3 humor scale. The winning teams (Morishita et al., 2020) combined the predictions of several models using sentence pair regression and ensembled the pre-trained language models BERT , GPT-2, (Radford et al., 2019) RoBERTa, (Liu et al., 2019) , XLNet (Yang et al., 2019) , Transformer-XL, and XLM (Dai et al., 2019) to form the final prediction.

Similar to humor detection, there has been some work to explore offense in text. SemEval '19 had a task (Zampieri et al., 2019), aimed at identifying and categorizing offensive language in social media. The top performing teams used ensembles of random forest, linear models, recurrent networks, and pretrained transformer language models.

Our work is motivated by the SemEval challenges which encourage interesting techniques to handle multiple word senses, cultural knowledge, and pragmatic competence. Through this challenge, we try to detect humor and explore the subjectivity of humor appreciation with a controversy score to examine the variance in humor ratings for each different text.

3 Dataset

3.1 Data

We use three types of datasets in this work: the HaHackathon competition dataset(Meaney et al., 2021), datasets for offensive text detection, and datasets for humor detection. We describe each of these in the following subsections.

3.1.1 HaHackathon Competition Dataset

The training dataset consists of 8000 texts (Meaney et al., 2021) and four subtasks: humor classification, humor rating, humor controversy, and offense rating. For our initial experiments we created a randomized 90-10 train-development split of 7200 training examples and 800 sentences for model development. In addition, the competition has its own development dataset of 1000 texts. The labels for this dataset were released during the final stage of the evaluation. The gold-standard test dataset (Meaney et al., 2021) is the one we use for our system results.

3.1.2 Humor Datasets

For the humor component of the competition, we use two datasets for intermediate finetuning: 200k Short Texts for Humor Detection (Annamoradnejad, 2020) and a self-compiled dataset of jokes and other texts scraped from Reddit.

The 200k Short Texts for Humor Detection dataset consists of 200,000 short text snippets, each labeled as either humorous or not humorous, with an even split between the two classes. The non-humorous texts are news headlines from the Huffington Post while the humorous texts were taken from Reddit communities such as /r/jokes and /r/cleanjokes.

The other dataset was one which we compiled ourselves, consisting of 200,000 snippets of text scraped from various reddit communities. This was primarily to address shortcomings we noticed in the 200k Short Texts dataset; namely the limited range of lengths for jokes and the singular source for the negative examples. For the positive examples of humor, we scraped the /r/jokes subreddit. For negative examples, we scraped subreddits such as /r/reddit.com and /r/worldnews, which offert more variability in the types of non-humorous texts than just news headlines from one news website.

3.1.3 Offense Datasets

The Offensive Language Identification Dataset (OLID) (Zampieri et al., 2019) is one dataset for identifying offensive language in texts, specifically tweets. It contains 14,200 tweets as well as binary annotations indicating whether or not they are offensive.

Another similar dataset is the Hate Speech and Offensive Language dataset (Davidson et al., 2017). This dataset consists of 26,953 tweets as well as labels corresponding to how many crowdflower users labeled them as hateful and/or offensive.

Together, we experiment with intermediate finetuning on both of these datasets in the hope that it would provide some benefit for offensive language detection in task 2.

4 System Overview

As a starting point, we use the HuggingFace Transformers library[2], along with their large collection of pretrained language models. From there, we finetune these transformers on the competition training dataset during development and the combined training and development datasets for the final evaluation phase.

Building on this basic paradigm, we experiment in four different ways with the goal of improving model performance: **(1)** using various datasets for intermediate finetuning, **(2)** using backtranslation to expand the competition datasets, **(3)** training multitask models to predict all labels simultaneously, and **(4)** ensembling predictions using different pretrained language models as starting points.

4.1 Intermediate Finetuning

We tried using intermediate finetuning on larger datasets for humor detection and offensive language detection in the hope that these larger datasets would provide a better starting point for training on the competition data, which is relatively small at just 8000 texts.

We perform intermediate finetuning in the same manner as the previously described basic transfer learning setup, and then we perform an additional transfer from the intermediate task to the competition task. For intermediate tasks we try using the two offensive language identification datasets previously mentioned, and for humor we tried using the 200k humor dataset and the Reddit dataset we collected.

We also try using ColBERT (Annamoradnejad, 2020), a pretrained BERT model that has been finetuned for humor prediction, as a starting point for intermediate finetuning and as a pretrained language model for the standard transfer approach.

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

4.2 Backtranslation Augmentation

As another method of expanding the training dataset and introducing variation, we use backtranslation to create paraphrases of the texts in the dataset. These paraphrases are generated by first translating the text into a different language using the Google Translate python library[3], and then translating the text back into the original language, usually with some small variations in the wording or sentence structure. This augmentation is useful in other tasks, but it was not clear whether the backtranslation would preserve humor, as some humor is generated based on the specific words or sounds used (e.g. puns).

4.3 Multitask Models

Initially, we train one model for each task or subtask. We also try training one model to learn to predict the labels associated with all four tasks or subtasks at the same time. To accomplish this, we attach four different heads on top of the final transformer outputs. Each prediction head consists of two fully-connected feed-forward layers matching the dimensionality of the transformer layers used by the pretrained language model, and an output layer that produces a single regression score or binary probabilities depending on the task.

4.4 Ensembling Model Predictions

We did most of our development with DistilBERT (unless otherwise specified) because it is relatively fast to train and run, allowing us to iterate more rapidly. We hypothesized that different pretrained language models would have different strengths and weaknesses when finetuned due to the different pretraining data used and the different model architectures. By ensembling (+Ens) many language models together, we might then counterbalance the weaknesses of individual models to improve overall performance.

Ultimately, we experimented with 6 model variants: "distilbert-base-uncased", "distilroberta-base", "bert-base-uncased", "roberta-base", "bert-large-uncased-whole-word-masking", and "roberta-large" pretrained language models from the HuggingFace Transformers library. To get the predictions for each model, we average together the predictions from 5 different random restarts to mitigate the effect of variance induced by the random initialization. To ensemble the different models

[3]https://pypi.org/project/googletrans/

together, we simply averaged the predictions from each model together to form the final predictions, taking the argmax of the averaged probabilities for classification tasks.

For a slightly more advanced ensembling method, for each task we select the models to average together by trying all possible combinations and selecting the combination that leads to the best performance on the development dataset (+Ens-Best). Then, we use the same model combinations to generate predictions to submit to the competition leaderboard.

4.5 Experimental Setup

To facilitate transfer learning on top of the original language model, we add two linear layers for each task on top of the CLS token of the transformer. The first linear layer has the same dimension as the language model. After the first linear layer, we use ReLU activation, and the second layer produces the prediction (classification or regression depending on the task). For training each model we use the same hyperparameters: a batch size of 10, a learning rate of 5e-5, 3 epochs, 500 warmup steps, and a weight decay of 0.01[4]. During intermediate finetuning, we transfer all weights from each prior finetuning step and all weights remain trainable at each step.

4.6 Results

We find that intermediate finetuning on other datasets for humor and offense identification do not improve performance. Similarly, using ColBERT as a starting point does not outperform other pretrained language models. This may be due to differences in how these datasets were sourced, and more analysis is provided in section 5. We also find that backtranslation augmentation is not helpful for humor detection, likely because it does not always preserve humor. While not beneficial, it is noteworthy as it is not clear whether prior work has explored backtranslation for expanding humor datasets, and this work suggests that backtranslation should not be used in contexts such as humor which are highly dependent on the specific words and sounds in a text.

Here, we show an example where backtranslation does not preserve humor since the word *imaginary* is a key part of the joke and it is substituted during the translation process:

- Original: My girlfriend is like the square root of -100. She's a 10 but she's imaginary.
- Backtranslation: My girlfriend is like the square root of -100. She is 10 but she is fantastic.

While many translations do not preserve humor, some translations do successfully preserve humor while introducing some word variation into the text:

- Original: My father doesn't trust anyone. In fact he has a saying... But he won't tell me.
- Backtranslation: My dad doesn't trust anyone he has a saying ... but he doesn't tell me.

Next, multitask learning improves performance over training models for each task individually, especially for some tasks such as humor rating prediction and humor controversy prediction. Finally, ensembling different pretrained language models together leads to an increase in performance, suggesting that these models complement each other by mitigating other models' weaknesses. Table 1 shows which submissions perform best on each individual task.

5 Error analysis

One of the possible sources of confusion and bias in our model seemed to be centered around atypical punctuation such as question marks and exclamation marks. For example, when a question mark was placed in the middle of a sentence, the model often erroneously labels it humorous regardless of the actual content. When manually reviewing the data, we found that the vast majority of texts that contain a mid-text question mark are humorous due to their setup and punchline structure. Without balancing with negative examples with similar structures, the model can become reliant on punctuation structure rather than the actual relationship between the words.

Another driver of error seems to be the actual source of the competition dataset. Through further analysis, we found that the vast majority of the negative examples seemed to be sourced from tweets. This can be seen in the length distribution of the dataset; there is a sharp cutoff around 140 characters, which used to be the maximum length for a tweet. However none of the other datasets we found or compiled ourselves (for Humor tasks)

[4]https://github.com/bzylich/humor-by-demographic

Approach	Humor F1 / Acc	Humor RMSE	Controversy F1 / Acc	Offense RMSE
RoBERTa-Large Multitask	**0.9510 / 0.9604**	**0.5339**	0.5220 / **0.4842**	0.4564
Multitask +Ens=6LM	0.9460 / 0.9565	0.5457	**0.5528** / 0.4841	0.4606
Multitask +Ens-Best=6LM	**0.9510 / 0.9604**	0.5411	0.5415 / 0.4659	**0.4530**

Table 1: Competition Results

contained information from Twitter specifically, almost all relied heavily on news headlines instead. When pulling individual examples, we found that tweets tended to use more colloquial language, with a greater variety of punctuation, vocabulary, and capitalization when compared to news headlines.

One final potential driver of error were song lyrics and quotes. There were a proportionally large number of movie, song, and TV show quotes used in the dataset, by a rough estimate based on sampling, approximately 5% of the example fell in one of those categories. Our model was often able to differentiate between these quotes, though it was not something that was found in our own custom datasets.

After performing this deep dive analysis on our results, and seeing the various areas of where our model got confused, we believe that the primary reason our models did worse with the inclusion of extra datasets was due to the source of our data. The wider range of punctuation, capitalization, and vocabulary expressed in twitter posts was not well captured by utilizing news headlines as a negative source, and thus likely allow our model to use syntax and punctuation structure as a substitute for the actual substance of the text.

6 Conclusion

In this competition, we explored the use of intermediate finetuning, backtranslation augmentation, multitask learning, and ensembling of different pretrained models. Unlike in many natural language processing tasks, intermediate finetuning on other related datasets provided no benefit in this task, perhaps because prior datasets used non-humorous texts that were much easier to identify. Next, although backtranslation augmentation did not improve performance, this is still a noteworthy result because it indicates that humor is likely not preserved through paraphrasing. Finally, multitask learning across humor and offense detection, as well as ensembling of different pretrained language models improved overall performance.

References

Issa Annamoradnejad. 2020. ColBERT: Using BERT sentence embedding for humor detection. *arXiv preprint arXiv:2004.12765*.

Rohan Badlani, Nishit Asnani, and Manan Rai. 2019. Disambiguating sentiment: An ensemble of humour, sarcasm, and hate speech features for sentiment classification. *W-NUT 2019*, page 337.

Fangyu Cai. 2019. Does ai get the joke?

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515.

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*.

David Donahue, Alexey Romanov, and Anna Rumshisky. 2017. HumorHawk at SemEval-2017 task 6: Mixing meaning and sound for humor recognition. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 98–102, Vancouver, Canada. Association for Computational Linguistics.

Nabil Hossain, John Krumm, Michael Gamon, and Henry Kautz. 2020. Semeval-2020 task 7: Assessing humor in edited news headlines. *arXiv preprint arXiv:2008.00304*.

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*.

Jihang Mao and Wanli Liu. 2019. A BERT-based approach for automatic humor detection and scoring. In *IberLEF@ SEPLN*, pages 197–202.

J. A. Meaney. 2020. Crossing the line: Where do demographic variables fit into humor detection? In

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pages 176–181.

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*.

Terufumi Morishita, Gaku Morio, Hiroaki Ozaki, and Toshinori Miyoshi. 2020. Hitachi at SemEval-2020 task 7: Stacking at scale with heterogeneous language models for humour recognition. In *14th International Workshop on Semantic Evaluations (SemEval-2020)*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*.

Orion Weller and Kevin Seppi. 2019. Humor detection: A transformer gets the last laugh. *arXiv preprint arXiv:1909.00252*.

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.

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.