The ColBERT Report

I. Abstract

Humor detection has a variety of interesting applications, from making virtual assistants more humanlike to capturing nuance in online reviews. While large, supervised models like BERT can perform quite well on this task, there is not a lot of labeled humor data available. Labeling humor data is expensive because humor is highly subjective, so labeling humor data is mostly done by humans. We found that using only 10,000 training examples with ColBERT could achieve an F1 score of 0.969. Our Baseline model, which simply passed pre-trained BERT embeddings into a final output layer, achieved an F1 score of 0.965 with 10,000 training data in half of ColBERT's runtime. While ColBERT successfully utilized linguistic structures of humor to achieve a high F1 score, BERT embeddings contributed to a lot of its success. When taking into account the tradeoff for runtime and model size, our Baseline model is a lot more efficient than ColBERT.

II. Introduction

As natural language processing has improved, researchers have turned towards harder tasks: recognizing non-literal language. An early humor detection study used Word2Vec combined with Human Centric Features (Yang et al, 2015), and since then most humor detection studies have used BERT variations, with a state-of-the-art F1 score of 0.982 at the time of writing. This is approaching difficult to beat for what has now become a simple classification task, but as students - one of whom is working on a 2015 MacBook Air - we wanted to know the simplest model trained on the least amount of data we could get away with and still achieve decent results: an F1 score in the high nineties.

Though each humor detection study proposes an elaborate architecture with increasingly larger datasets, we did not get significant improvements from adding layers on top of the base BERT model. In all cases but one we observed worse performance on the F1 metric compared to a baseline of one binary output layer on top of the BERT model, and the model that emulated the state-of-the-art only saw a 0.4% F1 score improvement but a significantly slower runtime. Additionally, we only saw a 2% decrease in our F1 score compared to the state-of-the-art while using 5% of their dataset to train on – a relief, since the resources just to tokenize the data were scarce. As a quantification of something subjective and nebulous, humor detection is uniquely positioned to capture the zeitgeist of many domains by helping to refine sentiment analysis, broaden a model's understanding of word meanings, and learn idioms and cultural references. Thus, we believe that a parsimonious humor detection model is valuable to academia and industry both.

III. Background

ColBERT, the current state-of-the-art model for humor detection, is a supervised binary classifier that takes a string as input, passes it through an eight-layer neural network that processes sentences in six parallel, non-sequential paths of inputs, and determines if the given text is humorous or not (Annamoradnejad & Zogh, 2021). Each path is essentially its own 2-layer neural network, and outputs from the paths are concatenated and fed into 3 more layers to output class probabilities. One of the paths takes in entire documents (ie. the entire short joke) as input, and the five other paths only look at individual sentences. For example, one path looks at first sentences in the documents, another path looks at second sentences in the documents, and so on.

Passing sentences in separately is an established technique for detecting incongruities between sentences, which the Script-based Semantic Theory of Humor (Raskin, 1985) asserts is the main mechanism for humor: contrast along an axis such as real/unreal, divine/mundane, possible/impossible, which is then connected with false analogy or some other device (West and Horovitz, 2019).

IV. Methods

ColBERT was able to achieve an F1 score of 0.982, but it was expensive. ColBERT was trained on 160,000 short texts, had 110,005,257 parameters, and took about one hour per epoch per 50,000 examples to train. The data tokenization process was also quite expensive, taking us one hour per 1,750 examples. Since acquiring labeled humor data is costly and data tokenization takes a long time, we explored ways to increase the efficiency of ColBERT by reducing the number of model parameters and training set size.

Data

We used Annamoradnejad & Zogh's dataset, a two hundred thousand document split evenly between humorous and non-humorous. Non-humorous examples were pulled from news headlines across several topics, and humorous examples were from the Reddit joke dataset, originally available on Kaggle. Annamoradnejad & Zogh were careful to balance average word length and other statistics across classes, and we preprocessed the data further by expanding contractions, spelling special characters out, or replacing them with English alphabet equivalents, e.g. "can't" => cannot, λ => "lambda," 🌮 => "taco-emoji", and ê => e.

Model Design

In order to feed the model individual sentences, we assumed that no document would have more than five sentences after we sent it through NLTK's sentence tokenizer. As a result, documents with more than five sentences were truncated, though this only impacted a tiny fraction of our dataset. We first created ColBERT_{10K}, which was replicated from the ColBERT model and trained on 10,000 short texts to test whether ColBERT could perform just as well with less data. Since we did not see other researchers compare their more involved models to a baseline, we

made a 1-layer model that outputted class probabilities using embeddings of whole documents as a way to test the usefulness of incorporating humor theory, since BERT is so successful at recognizing the most important words in both sentence and paragraph structures just on its own (Weller & Seppi, 2019).

We then proposed 4 other models with various input combinations to evaluate whether all components of the ColBERT model are essential to its high-achieving performance. These models are introduced in increasing complexity, with Baseline having the least number of parameters and ColBERT_{10K} having the most. All except ColBERT_{5K} were trained on 10,000 short texts:

- **ColBERT_{SENT}:** ColBERT model with 5 parallel paths, together representing the first 5 sentences of the documents as input. With ColBERT_{SENT}, we wanted to investigate whether detecting incongruity between sentences is sufficient for achieving the ColBERT model's high F1 score.
- **ColBERT_{DOC}:** ColBERT model with only whole documents as input. This model "turns off" the part of the ColBERT model used to identify incongruity between sentences and in doing so, helps to confirm that the humor theory did indeed play a role in the ColBERT model's success.
- **ColBERT_{SIMPLE}:** a simplified version of the ColBERT model with only 3 parallel paths of input, one for the first sentences, another for the second sentences, and the last for whole documents. Model 5 explores whether identifying incongruity between only the first two sentences is enough for ColBERT to beat the Baseline model.
- **ColBERT_{5K}:** ColBERT model trained on 5,000 short texts. Since, spoiler alert, the ColBERT model trained on 10,000 short texts (ColBERT_{10K}) performs quite well, we wanted to explore the lower bounds of the training size limit. This model has the same number of parameters as ColBERT_{10K}, our largest model.

We used 3,000 short texts as our validation set for all models. All models except Baseline used the adam optimizer with a learning rate of 0.0001, and Baseline used adam with a learning rate of 0.0005.

V. Results and discussion

None of our models beat the F1 score of the state-of-the-art ColBERT, but all of them beat the baseline test accuracy from guessing the dominant class for everything. ColBERT_{10K} achieved an F1 score of 0.969, the best F1 score out of all the models we compared. However, the Baseline model was just a fraction of a percent lower than ColBERT_{10K}, at 0.965. When we decreased the training set size to 5,000 short examples, the smaller ColBERT_{5K} model did not beat the Baseline model, though Baseline's F1 was merely 0.2% higher than ColBERT_{5K}'s. ColBERT_{SENT} performed the worst (F1 = 0.959), suggesting that learning incongruity between sentences alone is not sufficient for ColBERT to achieve high performance. Additionally, the fact that

ColBERT$_{DOC}$ performed better than ColBERT$_{SENT}$ indicates that while humor theory is an asset that distinguishes ColBERT from other high-performing models, it is more important for models to get contextualized embeddings of entire documents. Another explanation for ColBERT$_{SENT}$'s lower F1 is that ColBERT$_{SENT}$ truncates examples with more than five sentences, so it simply has less complete information than ColBERT did.

Figure 1 shows a detailed comparison of F1 scores across all models. All models had very similar F1 scores (± 0.6%) and performed better than the authors' best-performing baseline model, XLNet, which indicated that BERT embeddings played a huge role in ColBERT's success. We examined ColBERT$_{SIMPLE}$'s performance to evaluate whether it is crucial or redundant for ColBERT to have five sentence-level paths. We found that ColBERT$_{SIMPLE}$'s F1 score is lower than both Baseline and ColBERT, but the model's recall was the highest among all models. On the other hand, it has the worst precision out of all models. ColBERT$_{SIMPLE}$ is not very selective about what it considers humorous. From a practical perspective, this performance is not as desired since considering a piece of text to be humorous when it is meant to be taken seriously may generate a less desirable response than the other way around.

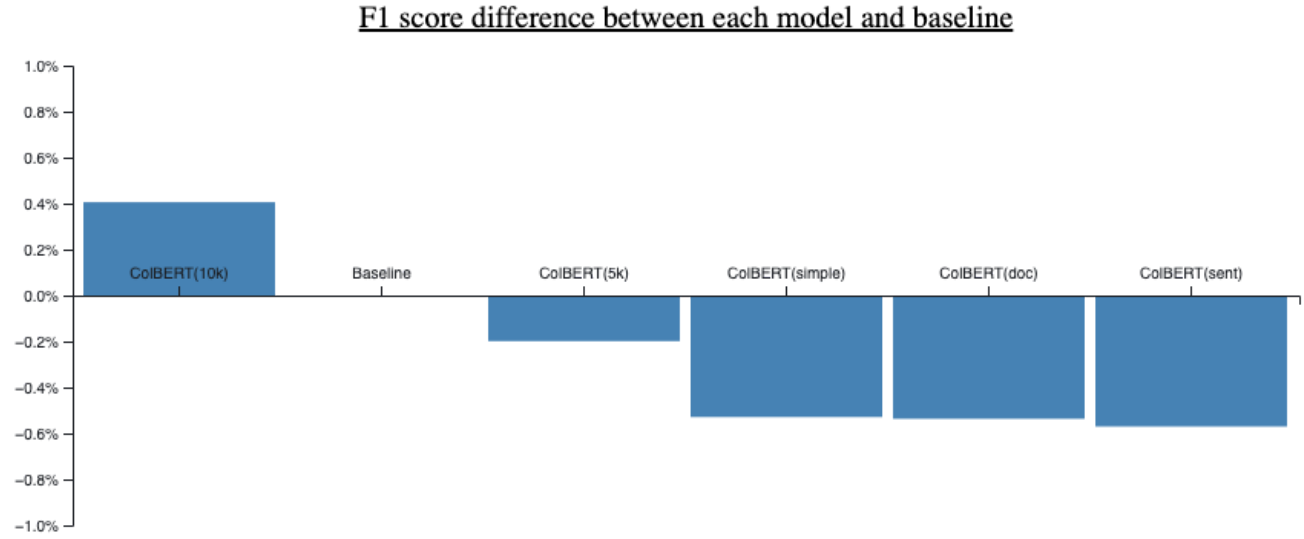

Figure 1. F1 score comparison between all of the models we created and the Baseline model. The Baseline model scored an F1 score of 0.965, and ColBERT had a F1 score of 0.982, which is 1.8% above baseline.

Our metrics for evaluating efficiency include model runtime and model size. Figure 2 shows the F1-runtime tradeoff. Our Baseline model is the model with the second highest F1 score and the lowest runtime out of all models. The Baseline model took 18 minutes to train 10,000 short texts, more than halving ColBERT$_{10K}$'s 45-minute runtime. When taking into account the runtime of the data tokenization process, the efficiency of the Baseline model is apparent. It took nearly 3 hours to generate 5,000 tokenized short texts for ColBERT, but only half of that time to generate tokenized data for the Baseline model. Compared to ColBERT$_{5K}$, the Baseline model still takes a shorter time overall to train. In addition to its shorter runtime, the Baseline model also has fewer

parameters than ColBERT. Baseline has 109,483,009 parameters, and ColBERT has 110,005,257 parameters. Despite being the smallest model, the Baseline model is still huge because it uses pre-trained BERT embeddings. Utilizing other embeddings could significantly reduce the model size, but as our exploration has shown, BERT embeddings are extremely powerful and hence worth using for the task of humor detection.

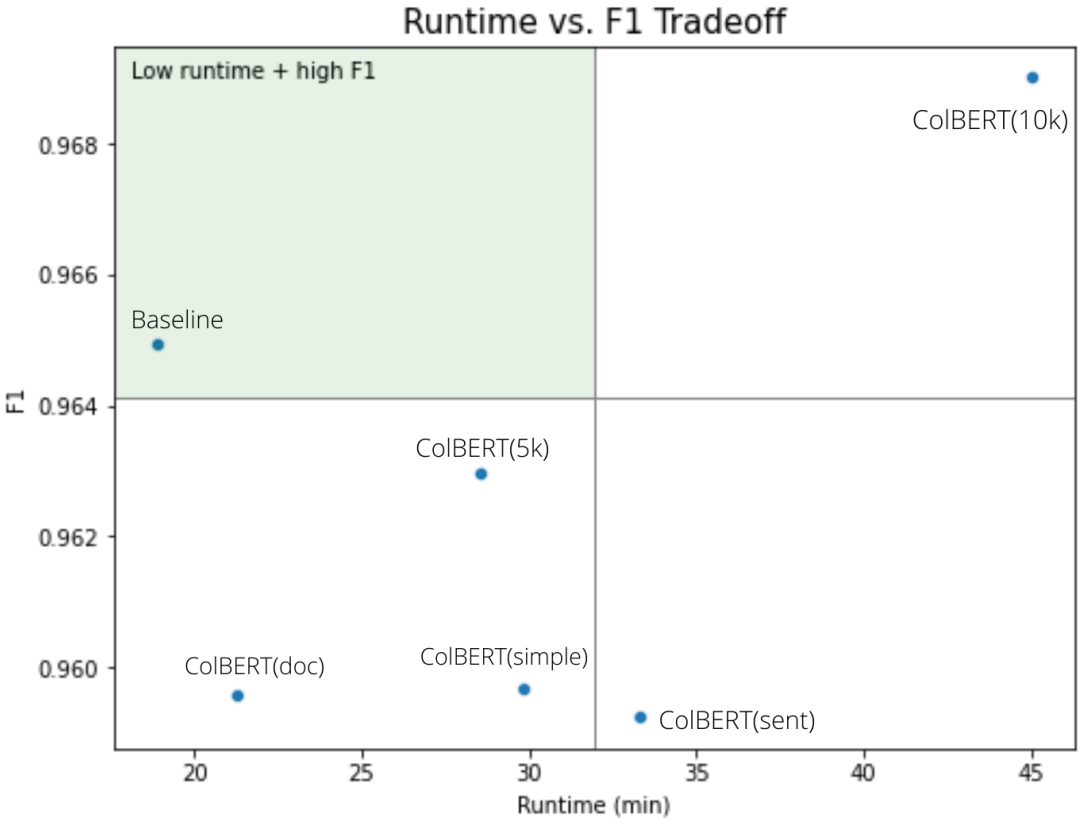

Figure 2. Baseline model is the only model that had a lower-than-average runtime and a high-than-average F1 score. In addition, it has the least number of parameters.

To understand where our scores came from, we looked at misclassified statements and saw that false negatives generally fell into three categories:

- They require world knowledge to understand, ie “Air bags: my car's attempt [at] cheering me up after accidents by giving me surprise balloons”
- The jokes contains words that the model might have most often seen in serious contexts, like “The boomerang is Australia's chief export (and then import)
- Puns/idioms, such as “Seize the day. Attack the week. Murder the month. Approach your life in a generally violent way.”

And false positives usually:

- Had dissonance or an unexpected element: “The easiest way to water plants is . . . using a turkey baster?”

- Had the choppy sentence structure that some jokes have, like “Dear lunch ladies, thank you. Sincerely, the parents.”

Additionally, Baseline correctly identified more single-sentence statements than ColBERT_{10K} did, suggesting that single sentences add noise to the neural net architecture. Or, since ColBERT_{DOC} (a neural net run on whole documents only) had a higher training accuracy but lower F1 score than Baseline, perhaps additional architecture can lead to overfitting in simple cases.

However, Baseline missed some jokes that had dissonance, like “God is pretty creative. I mean, look at me.” (divine/mundane). These things together suggest that ColBERT_{10K} does what Annamoradnejad and Zoghi suggest it does: capture inconsistency between sentences better than a whole-document embedding can. ColBERT_{10K} also seemed to capture irony, like in the “God is creative” joke, reasonably well. Irony is a class of humor that algorithms historically struggle to identify, making the recent NLP advances all the more impressive and also adding a consideration when choosing model complexity: if the use case needs to detect subtle humor, it might be worth the resources to use a larger neural net.

VI. Limitations and Future Directions

Given more time, we would have loved to explore model performance on a variety of training set sizes to better understand the tradeoff between F1 scores and resource consumption. Since sarcasm and irony is more difficult for a model to classify, it would also be interesting to explore transfer learning by testing our model on a dedicated sarcasm or irony dataset. Does what our model learned generalize to any non-literal statement, like idioms or figurative speech? Finally, we have some concerns about the dataset itself: it was scraped from the subreddit r/Jokes and posts with 200 or more upvotes were labeled as humorous. Reddit users are only a subset of the population, so our model may not generalize to other communities’ senses of humor, and we found a number of offensive jokes in the data as well. Finding offensive language is an entire subfield of its own and of course humor is subjective, but we would like to give more thought to what it means for a model to classify a statement as humorous. Is it classifying the statement as funny, or just non-literal? Does classifying an offensive joke as humorous condone the joke, or just recognize that such jokes exist?

VII. Conclusion

As expected, out-of-the-box BERT makes quality natural language processing accessible to those on a shoestring budget, but more complicated models trained on more data do boost F1 scores by a couple percentage points. In our research we did not see anyone compare a skeleton BERT model to one with a neural net on top, so we hope that our work shows a clearer benefit and tradeoff of using BERT embeddings to capture comedic dissonance.

References

[1] Issa Annamoradnejad and, Gohar Zogh, “ColBERT: Using BERT Sentence Embedding for Humor Detection” in arXiv:2004.12765v5 [cs.CL], 2021

[2] Orion Weller and Kevin Seppi, “Humor Detection: A Transformer Gets the Last Laugh” in arXiv:1909.00252v1 [cs.CL], 2019

[3] Raskin, V. (1985). Semantic Theory of Humor. In: Semantic Mechanisms of Humor. Synthese Language Library, vol 24. Springer, Dordrecht. https://doi.org/10.1007/978-94-009-6472-3_4

[4] Diyi Yang, Alon Lavie, Chris Dyer, and Eduard Hovy, “Humor Recognition and Humor Anchor Extraction” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 2367–2376.