Sarcasm Detection in Twitter Data

Anan Aramthanapon
Harvey Mudd College
aaramthanpon@hmc.edu

Saatvik Sejpal
Harvey Mudd College
ssejpal@hmc.edu

1 Introduction

We propose the following Research Question: Given a dataset of a few thousand English language Tweets, how accurately can we predict whether or not a particular Tweet employs sarcasm? Furthermore, does the accuracy of our model improve if we train it on some context for the Tweet in question?

The dataset we intend to use for this task is a set of Twitter Data from FigLang 2020's Shared Task. This dataset has the context for the tweet (the preceding comment), the potentially sarcastic response, and a label denoting whether or not a Tweet is sarcastic.

To answer the Research Question, we propose to explore using models such as Bag of Words with tf-idf and n-grams with tf-idf, and then use classifiers such as NaiveBayes to train and test on the data. We will also use Deep Learning based models such as BERT (Devlin et al., 2019) as used in (Ghosh et al., 2020) for a very similar task but on a different dataset. Such models have proven to be very effective for sarcasm detection in the past (Ghosh et al., 2020). Additionally, we will also explore the effect of context on sarcasm detection by providing the preceding text and the sarcastic response, and comparing the results of providing the preceding text to results without.

2 Literature Review

2.1 Related Work

The experiment performed in Abu Farha et al., 2021 An experiment similar to what we are proposing to do using a BERT encoder has been done in the past (Abu Farha et al., 2021), but it is different in that it also combines the features of sentiment analysis in the model, and is done in the Arabic language. This is slightly more complex than what we are proposing to do, but there is a lot of useful information about using BERT for a sarcasm task from this paper. We will also use n-grams for our experiment as n-grams are a common feature used in sentiment analysis tasks, and this paper has shown that sentiment analysis features are helpful for detecting sarcasm.

A summary of sarcasm and irony detection as well as a subtask of breaking down and specifying the type of sarcasm and irony is available for SemEval-2018 Task 3 (Van Hee et al., 2018). Our task is similar to this but will only tackle the main task of detecting sarcasm without the sub classifications. This is because the results of the paper show that it is difficult to detect the specific type of sarcasm detected at this point. Instead, we will further investigate whether using models such as Bag of Words and tf-idf or n-grams will help improve the model's performance, as this paper only discussed the overall results of the top-performing teams for the task which mostly used deep learning techniques to perform the classifications. We will compare how using basic NLP techniques compare to using BERT.

There has been research on exploring the task of Sarcasm Detection with the help of context (Ghosh et al., 2020), (Hazarika et al., 2018). One paper discussed modeling conversation context in Twitter Data/Discussion Forums, and found that using conversation context helps with sarcasm detection (Ghosh et al., 2020). The paper makes use of models such as LSTMs with attention to understand context and make predictions. A different paper uses a CNN along with user profiling to provide the model with more context on the corpora (Hazarika et al., 2018). The method we are proposing does not do user profiling, but we will explore whether the added context of the preceding comment before the sarcastic comment will help models better detect sarcasm.

Sarcasm is challenging to label because intended

sarcasm is not always the same as perceived sarcasm (Oprea and Magdy, 2019). Because of this, even hand-labeled datasets for sarcasm are likely to be noisy as readers may perceive a text in a way that does not agree with the author's intentions. Additionally, automatically tagging sarcastic tweets through certain keywords such as "#sarcasm" is not effective because this has also been shown to be very noisy and this also fails to capture many types of sarcasm (Oprea and Magdy, 2019). Because of this, we have decided to use a dataset which contains sarcasm labels by the authors themselves.

2.2 Methods

We will use the aforementioned dataset for this task (Ghosh et al., 2020), which contains 5000 datapoints, each containing the preceding comment and the potentially sarcastic response pulled from Twitter, as well as the ground truth label for whether the response is sarcastic or not.

We plan to answer our research question by performing a series of experiments. Our first baseline experiment will be to use a model based on Bag Of Words along with tf-idf (Salton and McGill, 1986) using the Naive Bayes classifier, which is a baseline also used in similar experiments in the past (Hazarika et al., 2018). We will also experiment with a model based on n-grams and tf-idf. We will compare how these baseline models compare to a state-of-the-art deep learning BERT model (Devlin et al., 2019) for the same task. For each model, we will conduct two experiments: one with the preceding comment and the response, and one with just the response to see the impact context has on sarcasm detection.

Before we can perform any experiments or analysis, we need to process our data by splitting our training data into training and validation data. For our experiments, we decided to hold out 20% of our training set as a validation set. After this, we convert our data into Tensors so that we can run models like BERT on it.

For our deep learning model, we plan to use a pre-trained BERT model. For this process, we plan to use the pre-trained BERT models included with Pytorch's Transformer library. More specifically, we plan to use bert-base-cased and bert-base-uncased to compare how capitalization plays a role in detecting sarcasm. For each model, we will use the tokenizer shipped with the BERT model we are using.

We hope to gain insights about two main aspects:

- How does using context affect the accuracy of our model?
- How does using each of the different models affect the accuracy of sarcasm detection?

By having one model that uses context, and another model that does not use context, we are able to determine if having context actually impacts the accuracy of our models.

To evaluate the models, we will use the standard metrics such as accuracy, precision, and F1 score.

To help visualize our results, we will have one figure that displays the words that show up most frequently in tweets that are categorized as sarcastic by our model, excluding stop words. We will have another figure that summarizes the differences in our models' performances with and without context, as well as using our baseline n-gram model against BERT. We will also add a table to discuss certain testing examples for which our models had the worst performance, and try to analyze why our model didn't do well on those specific cases. We will also compare these results between models to see if there are certain features in some types of sarcasm that make them hard to detect in general or if there are certain areas in which a certain model is not effective at predicting.

References

Ibrahim Abu Farha, Wajdi Zaghouani, and Walid Magdy. 2021. Overview of the wanlp 2021 shared task on sarcasm and sentiment detection in arabic. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 296–305. Association for Computational Linguistics (ACL). The Sixth Arabic Natural Language Processing Workshop, WANLP 2021 ; Conference date: 19-04-2021 Through 19-04-2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Debanjan Ghosh, Avijit Vajpayee, and Smaranda Muresan. 2020. A report on the 2020 sarcasm detection shared task. *CoRR*, abs/2005.05814.

Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. 2018. CASCADE: contextual sarcasm detection in online discussion forums. *CoRR*, abs/1805.06413.

Silviu Oprea and Walid Magdy. 2019. Exploring author context for detecting intended vs perceived sarcasm. *CoRR*, abs/1910.11932.

G. Salton and M. J. McGill. 1986. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. SemEval-2018 task 3: Irony detection in English tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana. Association for Computational Linguistics.