Predicting Heursitscs Related to the Controversiality of a Social Media Post

Henry Tischler, Gene Huntley - Team Members

Jenifer Hooten - Sponsoring Teacher

Team #10

Academy for Technology and the Classics

April 2023

Contents

1 Executive Summary

Social media platforms are currently confronted with a large amount of controversial content, which carries significant consequences for the emotional health of their users [1]. Without an effective way to classify this content, social media platforms face difficulty when trying to limit such content.

In this project, we attempt to use deep learning to predict the metrics (heuristics) that a social media platform would use to categorize a post as controversial before the post is even reacted to - simply from the natural language content of the post. By doing this, a post can be classified as controversial, before it's already had a negative impact on a social media platform.

To achieve this, we made use of a BERT model, with our custom layer trained to identify content as controversial. With this, we were able to make use of a pre-trained understanding of natural language, while still creating a model relevant to our task, achieving great success on what would otherwise be a limited dataset.

Through our model, we were able to achieve a high degree of accuracy, successfully demonstrating the potential of our approach in detecting controversial content, and allowing this detection to happen much quicker than otherwise possible.

2 Project Problem and Solution

As nearly any frequent user of a social media platform can attest to, most social media platforms contain a large amount of controversial content. The constant exposure to such content, for many, can harm their emotional health [1].

To reduce the negative impact of controversial content, it may be useful to reduce one's exposure to this content. However, to reduce the exposure of controversial content, it is of course necessary to have the capability to classify content as controversial or non-controversial.

Controversial content is relatively easy to recognize after it's been published, by analyzing how the content is being reacted to. However, this of course has its limitations, as the content has to have a negative effect before it's controversiality can be recognized.

In this project, we use deep learning techniques to predict heuristics related to the controversiality of a social media post, simply from its natural language content. We do this by learning the features of the natural language which correspond with a post's controversiality.

3 Model Structure

To effectively work with natural language, we opted to use a deep learning model. This model is able to take the textual content from a variety of social media posts and learn what within this text content correlates with a controversial post.

3.1 Use of Heuristics

A heuristic is, essentially, a metric that itself may not correlate exactly with what needs to be predicted, but can be used as a part of an informed judgment.

In this project, we make use of heuristics related to post controversiality. Specifically, we use the like-to-retweet ratio of a Twitter post, though other similar metrics exist. These metrics, in many cases, can give us an idea of the controversiality of a post. For example, if a post has many retweets, though not many likes, there's a possibility that users felt the need to give their thoughts on a post, instead of simply expressing their agreement. Thus, a low like-to-retweet ratio is possibly an indicator of controversial content.

Unfortunately, without a large annotated dataset, it's impossible to validate if these heuristics truly correlate with the underlying controversiality of a post. However, while we cannot, in this project, predict controversially itself, by predicting these metrics using nothing but the content of a Tweet, we can demonstrate the potential of using these heuristics to predict the controversiality of a social media post ahead of time, avoiding the need to inflict the negative impact of such content before it can be categorized.

Additionally, while some work has been done in the past into the training of models which use annotated data, these studies often suffer from the relatively low sample size afforded by manual annotation [4].

3.2 BERT Modeling

The simple way to view BERT modeling is as a way to leverage a deep understanding of natural language, from an extremely large, pre-trained model, and then narrow the capabilities of the model to exactly your task, with relatively minimal training.

Essentially, the BERT model makes up the first layers of the neural network. We can then add our layers onto the end of that neural network, and train those layers to our specific task. This way, our model doesn't have to learn about the English language itself and can focus entirely on our specific task.

To provide some more technical detail, "BERT" stands for "Bidirectional Encoder Representations from Transformers". The model can successfully consider the bidirectional context of a word. That is - instead of just providing an idea of what a word means in its most general context, such as what's done with a word embedding, or providing context for a word based on the words preceding it. [2] Instead, BERT can effectively consider the context of a word within the sentence it's in, providing a very rich and useful understanding of natural language [3], which can be applied to a narrower context.

3.3 Our Own Layer

The BERT model we are using, of course, doesn't have any understanding of how to associate the natural language content of a social media post with controversiality. To do this, we needed to add our custom layer, which takes input from the output neurons of the BERT model and outputs its predictions of the metrics which we use to approximate controversiality.

For the primary metric used, like retweet ratio, we created ten different classes for the metric, each of which represents a percentile range of the metric across our training dataset.

So, for example, the first class would represent all like/retweet ratios in the bottom [0-10%), the second class would represent all like/retweet ratios in the [10%-20%) percentile range, and so on.

By using these percentiles, we can make our classes very close to equal in presence, certainly within our dataset, and gain a relatively detailed idea of the predicted metric.

4 Dataset Collection

To train our model, we, of course, need a dataset to use. For this project, we were unable to find a pre-existing, publicly available dataset that would work well for this project. For our dataset, we had three main constraints.

1. The dataset needs to be as large as possible, to allow for the most effective training of our network.

2. The dataset needs to have as much information as possible relating to the public reactions to a tweet.

3. The dataset needs to consist of tweets that have been reacted to as widely as possible, to provide the richest possible reaction metrics.

To fill these constraints, we collected a dataset of Tweets, using a Python program written ourselves. This program aggregated the tweets from the top 50 Twitter accounts, by using the Twitter API to scrape these Tweets over a relatively long period, and compiled all of these Tweets into a consolidated dataset.

In this dataset, we included every possible metric which we could relate to the public reaction to a Tweet. In the end, we used three metrics to approximate the controversially of a post, the first metric is the primary one utilized in this experiment.

In total, we were able to collect 156,677 tweets, each of which was used to train the neural network.

5 Solution Validation Methodology

To validate that our solution performs effectively, it's necessary to perform a test of our model. The testing of our model, following extremely standard Machine Learning methodology, relies on testing the accuracy of our model on a validation dataset. This dataset would not have been shown to the model during training. As such, testing the accuracy of our model on a validation

dataset provides a very close approximation of the performance of the model on data it will encounter when deployed.

In our case, the validation dataset used was taken from our total dataset, representing 25% of our entire dataset.

To gauge the accuracy of our model, we used two different metrics. The first, categorical accuracy, is a fairly standard accuracy metric in the field of machine learning. For this model, this simply represents the accuracy with which the exact category is predicted.

However, unlike most classification models, with our particular model, not all misclassifications are equally incorrect. A misclassification in a nearby percentile should be represented as being more accurate than one in a further percentile. As a result, we also used the "average separation" metric, which represents the average distance between the selected percentile and the correct one.

6 Solution Validation Results

Our model, after three epochs of training, was able to achieve a validation accuracy of *29.94%*, meaning that 29.94% of the posts were categorized into the correct percentile. While not perfect, this level of accuracy still gives a strong indication of the potential in the model, performing much better than the *10%* accuracy that would be archived without any learning. Additionally, the trend from the three epochs of training, seen in Figure 1, seemed to be fairly linear, showing the potential for this accuracy to improve should more compute

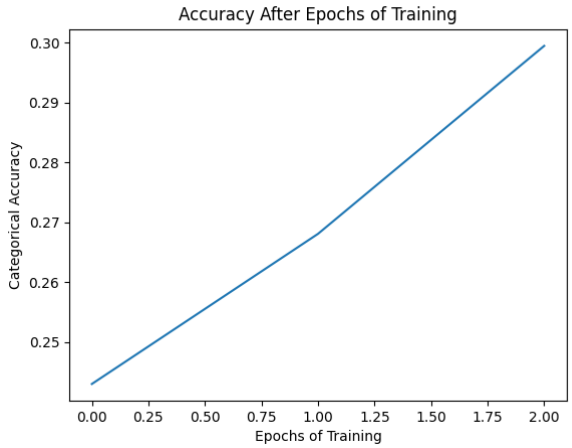

Figure 1: A graph showing the categorical validation accuracy of our model as training progressed.

resources be invested into training the model.

Additionally, we were able to achieve an average separation of 20.7%. This means that, on average, the percentile assigned to each post deviated from the ideal percentile by 20.7%. This, while of course not precisely accurate, shows that this model can be used to give a relatively accurate general idea of how controversial a post is, even when it fails to identify a post's controversiality exactly.

7 Conclusions

In this project, we were able to successfully teach a machine learning model to recognize the signs of controversial content and achieve a level of accuracy that, while not perfect, still allows for the model to be useful in the detection

of controversial content. Through the use of natural language processing, we have shown that it's possible to predict the metrics which can indicate controversiality. Through further work and exploration into different natural language processing techniques, it's possible that the accuracy could be improved even further.

8 Future Work

To expand on this project, given more time and resources, there are several improvements and expansions which could be made. The largest improvement to this project itself would be to train the BERT model for more epochs, to ascertain a more complete idea of the peak performance that can be archived. Additionally, collecting more data from Twitter, over a larger period, would allow for more performance to be achieved. And, of course, the dataset could be expanded to other forms of context, such as spoken language or pictures with text. Finally, given the resources to annotate enough content, the correlation between the statistics

9 Acknowledgements

We would like to provide the following acknowledgments to the individuals and organizations who assisted us with this project:

1. Our Teacher and sponsor, Ms. Hooten, for her support and facilitation of the challenge of the year.

2. Nicholas Kutac for Reviewing our Interm Report and Providing Feedback on the Project

3. The Supercomputing Challenge for facilitating the challenge and providing the associated support with the project.

References

[1] William J. Brady et al. "How social learning amplifies moral outrage expression in online social networks". In: *Science Advances* 7.33 (2021), eabe5641. DOI: 10.1126/sciadv.abe5641. eprint: https://www.science.org/doi/pdf/10.1126/sciadv.abe5641. URL: https://www.science.org/doi/abs/10.1126/sciadv.abe5641.

[2] Jacob Devlin and Ming-Wei Chang. "Open Sourcing BERT: State-of-the-Art Pre-training for Natural Language Processing". In: (2018). URL: https://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html.

[3] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL].

[4] Marzieh Mozafari, Reza Farahbakhsh, and Noël Crespi. "Hate speech detection and racial bias mitigation in social media based on BERT model". In: *PLoS ONE* 15 (2020).