Elliot Tan

CS522 Term Project – Final Report

Evaluating the effectiveness of GPT-2 generated hate speech for SVM and BERT hate speech classification

Group members: Anjali Veer, Alexis Edwards, Akhilesh Datar

12 May 2021

Introduction

Abstract

The aim is to see if using GPT-2 generated data to train a predictive hate speech model will improve the accuracy of the model. The models we will be building are from using SVM and

BERT. We will be comparing the effects of different amounts of GPT-2 generated data on the accuracy of the model. This project will be completed in python and run on google Colab.

1 Goals and Methodology

1.1 Goals and hypothesis

The goal of this project is to determine the effects of using GPT2 generated training data on the accuracy of hate speech detection.

Hypothesis: Adding GPT2-generated data to our classification experiments will decrease the accuracy of our SVM and BERT classifications significantly, perhaps by ten to twenty percent.

1.2 Methodology

We will use a training data set from Kaggle that has 25000+ rows of data. We will partition some of it for training, and the rest will be left untouched except for testing. We will first see how increasing the amount of training data will affect the test result accuracy. Then we will take the lowest amount of training data and use GPT-2 to generate more data. We will then use the combination of the original dataset together with the GPT-2 data and train a model. We will then see how accurate that model is.

We will not only be comparing how accurate the generate data is to the original data, we will also be looking at how the accuracy of the model changes with the amount of generated data

2 Generative Models

Generative language models generate text based on the theme or idea of the phrase that is given as the input. GPT-2 is an AI transformer that implements deep neural networks to determine human-like output given a certain input. We will be using this model on tweets containing hate speech and other forms of tweets to generate training data.

Data and Computational Setup

2.1 Data Sources

We will be using data from Kaggle at this link: https://www.kaggle.com/mrmorj/hate-speech-and-offensive-language-dataset.

This data set has 7 columns: index, count, hate_speech, offensive, neither, class, tweet.

There is a total of 25296 rows in the original data set, but we plan to only use a small portion of it for training and testing. The size of the data is about 2.42 megabytes.

2.2 Data Format

The only columns we are currently interested in are ‘class’ and ‘tweet’. The class is the label that categorizes whether the corresponding tweet is hate speech, offensive language, or neither.

2.3 Programming Language, Computational Resources

We will be using Python3 programming language. We will be using the following packages: pandas, io, sklearn, numpy, tensorflow, transformers, torch, warnings, Bert
In order to gain the computational power needed, we will be using google Colab to run our python scripts.

Experiments

1. Baseline Experiments
   A. The baseline experiments consist using 200, 500, 1000, 2000 rows of the original data set and doing it both with and without pre-processing for SVM and BERT
   B. For pre-processing the tweets removed consist of:
      - Tokens beginning with the '@' symbol to get rid of twitter usernames
      - Links
      - Dates and times
      - Emojis (which were converted to symbol text)
      - Random symbols (noise)

      SVM: A tf-idf was used for SVM word vectorization with maximum features to 5000. This means that the maximum possible number of unique words was 5000.

      BERT: Word to word tokenization was performed on the twitter text, as well as one hot encoding. A basic softmax layer was put on the base BERT model as well. In order to maximize the training and testing scores, a basic hyperparameter model using optimizers from transformers AdamW was used. We ran this model over 10 epochs.

2. Bert and SVM training
   SVM
   A. Experiments were run on SVM's defaults using 5000 maximum features.
   B. A final set of experiments was run on the post-processed set of data with a tuned kernel type
      a. Experimented with a polynomial kernel, sigmoid kernel, and default rbf kernel
      b. Affects the shape of the line/curve separating the classes.

   BERT
   A. Most experiments were run on default BERT features.
   B. Utilized AdamW weight decay optimizer model as a hyperparameter.
   C. Used base 0.3 dropout rate for dropping redundancy

3. Experiments with generated data

A. The same experiments as the baseline were conducted, except for experiments with over 200 rows, All tweets after the initial 200 were generated by a sample of the original entire dataset.

B. Experiments for both SVM and BERT were done over datasets that were

   a. Without pre or post processing

   b. With pre without post processing

   c. With pre and post processing

C. The labels for the generated data were simply a copy of the label that the original tweet had been labelled as.

D. The original tweet used to generate the GPT-2 data string was removed from the entire string that was produced, leaving purely GPT-2 generated tweets

Generated Data

4. Quality of generated data

   A. The first set of GPT-2 generated data was noisy because of Twitter usernames and links. Many tweets did not contain relevant content.

   B. The pre-processed set had less noise, but did not reflect the same sentiments as the original data. This could be due to a mistake in the generation algorithm not feeding enough input.

   C. The data generated was generally irrelevant to hatespeech.

   D. The final set was significantly improved from post processing. It was more likely to resemble the sentiment of the given input and the generated text more closely resembled the given input

5. Post processing

   A. The same function used in pre-processing was used for post-processing.

6. Analysis of generated records

   A. A sample of 100 rows of a set of 1800 generated data from pre-processed original text was taken and examined row by row to determine if each row was labelled correctly according to what had been generated. It was found that 20 rows of the sample were labelled as hate speech, and 80 were not. Of the 20 hate speech, we found 10 to be mislabelled. Of the 80 non hate speech, 2 were mislabelled. If this sample is taken to be an accurate picture of the generated data, it would mean that the GPT2 generator is very accurate at producing non hate speech data when given non hate speech input, but cannot consistently produce hate speech data when given hate speech input.

      It was found that many of the incorrect labels of hate speech were due to the fact that although the text generated had offensive language, the sentiment was not one necessarily of hate.

Results

Without Pre/Post-Processing

Experiments - Baseline

	SVM			BERT			
Total Training Size	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Performed By**
200	0.72	0.53	0.47	0.79	0.82	0.80	Alexis
500	0.82	0.60	0.61	0.88	0.82	0.84	Anjali
1000	0.88	0.75	0.80	0.91	0.88	0.88	Elliot
2000	0.88	0.75	0.80	0.93	0.91	0.92	Akhilesh

Experiments - GPT2

Data			SVM			BERT			
Total Training Size	**Original**	**GPT2-Generated**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Performed By**
200	200	0	0.72	0.53	0.47	0.79	0.82	0.80	Alexis
500	200	300	0.81	0.57	0.57	0.82	0.84	0.83	Anjali
1000	200	800	0.83	0.6	0.62	0.88	0.7	0.75	Elliot
2000	200	1800	0.86	0.7	0.73	0.87	0.68	0.72	Akhilesh

Pre-Processing

Experiments - Baseline

	SVM			BERT			
Total Training Size	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Performed By**
200	0.79	0.5	0.44	0.81	0.77	0.78	Alexis
500	0.79	0.5	0.44	0.86	0.85	0.86	Anjali
1000	0.81	0.57	0.57	0.86	0.82	0.84	Elliot
2000	0.84	0.66	0.7	0.91	0.87	0.89	Akhilesh

Experiments - GPT2

Data			SVM			BERT			
Total Training Size	**Original**	**GPT2-Generated**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Performed By**
200	200	0	0.79	0.5	0.44	0.81	0.77	0.78	Alexis
500	200	300	0.79	0.51	0.47	0.85	0.77	0.8	Anjali
1000	200	800	0.81	0.55	0.53	0.9	0.74	0.79	Elliot
2000	200	1800	0.81	0.57	0.57	0.93	0.79	0.84	Akhilesh

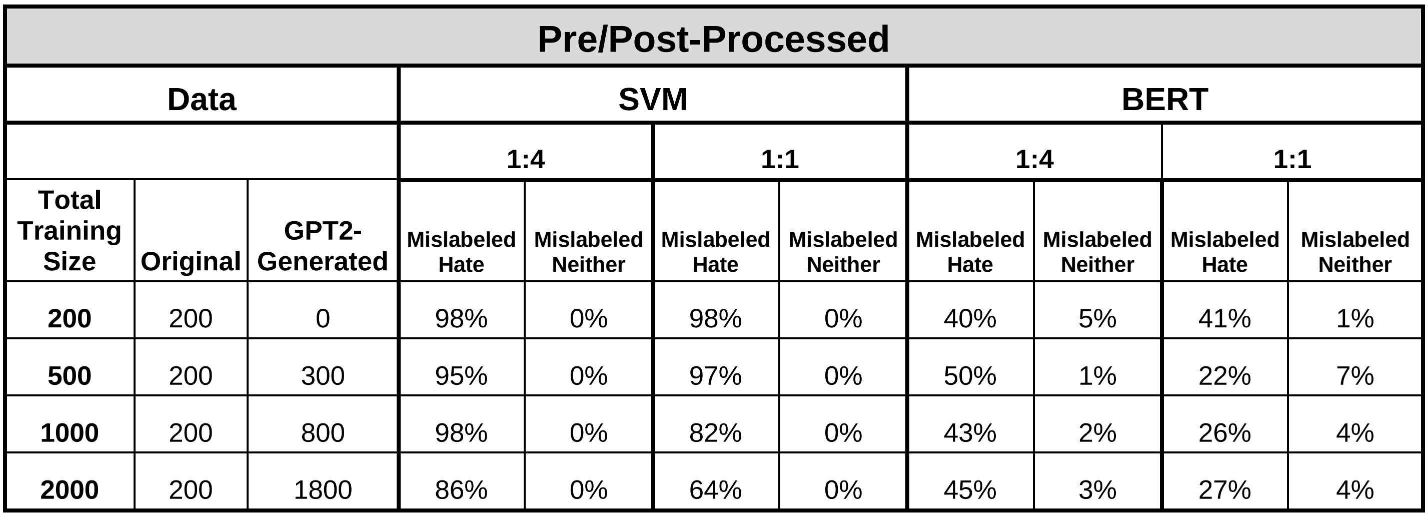
Pre/Post-Processing

Experiments - GPT2

Data			SVM			BERT			
Total Training Size	**Original**	**GPT2-Generated**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Performed By**
200	200	0	0.79	0.5	0.44	0.9	0.7	0.73	Alexis
500	200	300	0.79	0.53	0.49	0.86	0.74	0.76	Anjali
1000	200	800	0.81	0.55	0.53	0.82	0.8	0.81	Elliot
2000	200	1800	0.82	0.6	0.61	0.85	0.77	0.8	Akhilesh

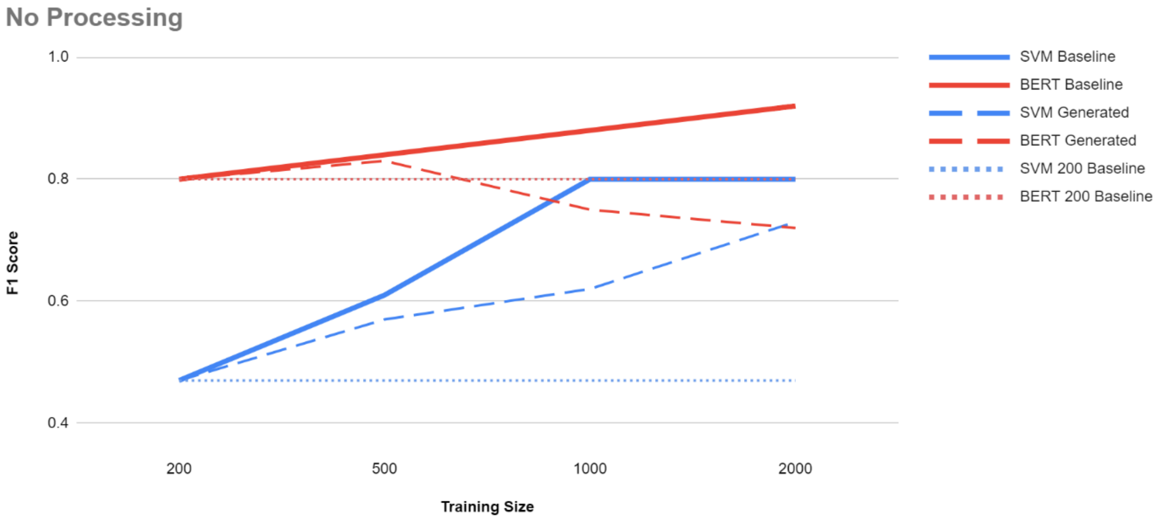
SVM Parameter Tuning: Kernel

Data			rbf (Default)			poly			sigmoid		
Total Training Size	**Original**	**GPT2-Generated**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
200	200	0	0.79	0.5	0.44	0.79	0.51	0.47	0.84	0.64	0.68
500	200	300	0.79	0.53	0.49	0.79	0.51	0.47	0.85	0.68	0.71
1000	200	800	0.81	0.55	0.53	0.79	0.51	0.47	0.85	0.67	0.71
2000	200	1800	0.82	0.6	0.61	0.79	0.51	0.47	0.86	0.69	0.73

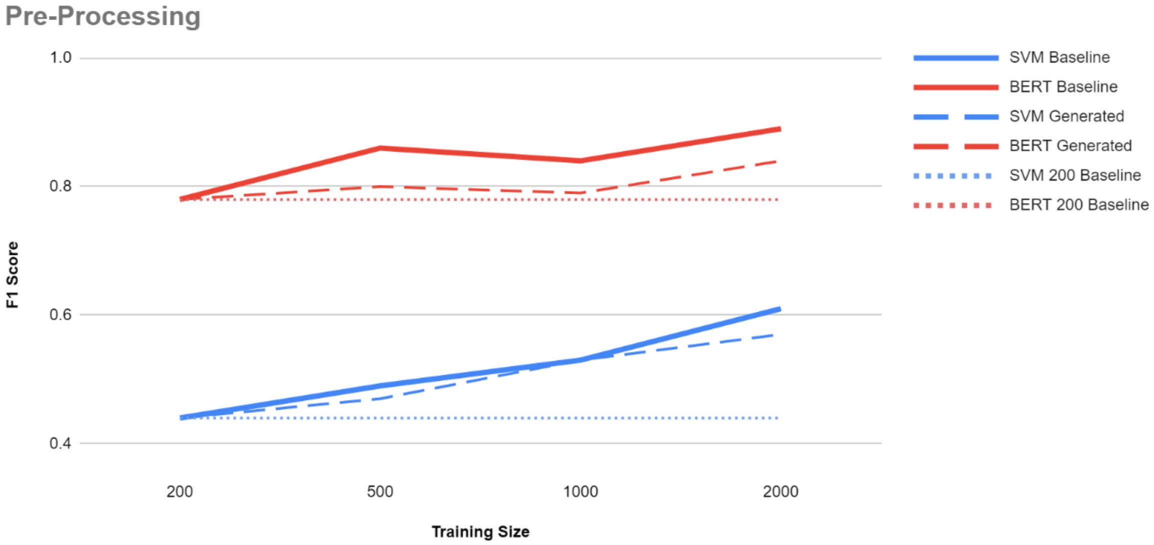


Pre/Post-Processed										
Data			SVM				BERT			
			1:4		1:1		1:4		1:1	
Total Training Size	Original	GPT2-Generated	Mislabeled Hate	Mislabeled Neither	Mislabeled Hate	Mislabeled Neither	Mislabeled Hate	Mislabeled Neither	Mislabeled Hate	Mislabeled Neither
200	200	0	98%	0%	98%	0%	40%	5%	41%	1%
500	200	300	95%	0%	97%	0%	50%	1%	22%	7%
1000	200	800	98%	0%	82%	0%	43%	2%	26%	4%
2000	200	1800	86%	0%	64%	0%	45%	3%	27%	4%

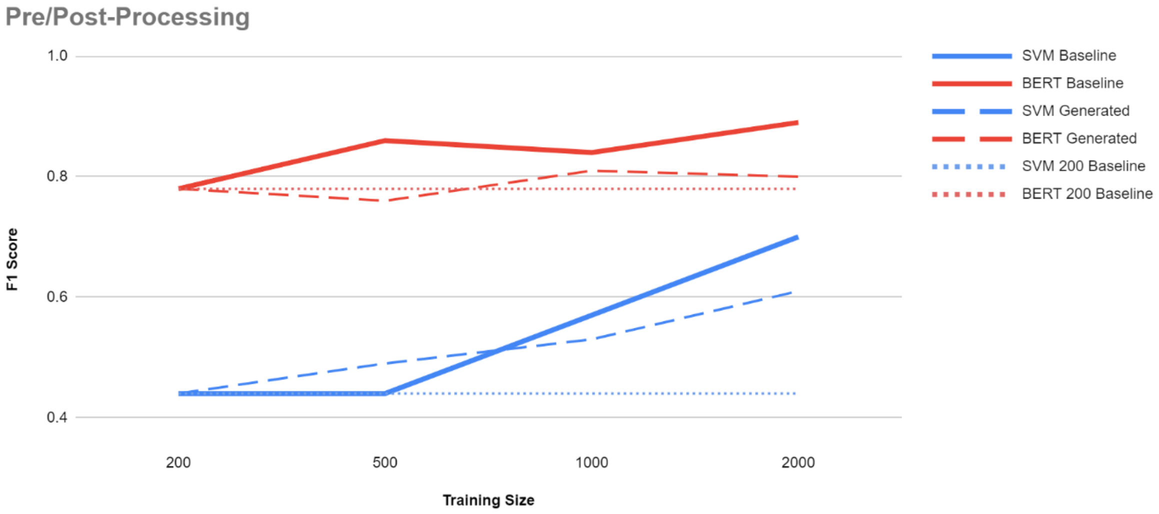
Labelling accuracy



Graph 1: no processing



Graph 2: pre processing



Graph 3: pre and post processing

Analysis and Conclusions

a. Evaluating effectiveness of GPT-2 generated data:
From the data above, we conclude that GPT-2 data can be a viable method of creating training data to train models for hate speech classification given that the inputs used for GPT-2 are properly processed.

- Looking at graph 1, we see that when feeding unprocessed data as input into GPT-2, it is clear that the GPT-2 data does not perform well. If we look at the f1 scores for no processing for SVM in comparison to real data, we see that it is lower in accuracy by about 10%. If we look at the f1 scores for BERT, we see a decrease in accuracy as the the amount of GPT-2 data increases. From this, we would not recommend using raw unprocessed input for GPT-2 text generation, as it would lower the accuracy of your model.
- Looking at graphs 2 and 3, both of which use processed inputs for GPT-2 generation, we see comparable results between the real training data and GPT-2 generated data. Although the real data still out performs the generated data, both see an increase in the f1 score as the amount of training data increases. For example, we see large improvements in the f1 score for SVM at 2000 rows (200 real data, 1800 generated data), improving the classifier by about 10% to 15% across all experiments. We also see slight improvements in the f1 score for BERT at 2000 rows (200 real data, 1800 generated data) of about 1% to 10%. This suggests that the GPT-2 data was effective in training a model to recognize hate speech. The only exception to this is BERT model using generated data at 500 rows (200 real data, 300 generated data). However, we would not say this is significant enough to show that GPT-2 data is not effective, and could be counted as an outlier.

b. Comparing effectiveness using different test set distributions:

- After closer examination of our first test set (1:4 ratio of hate speech: non hate speech), we realized that both the SVM and BERT models were excellent at classifying non hate speech (all the 1:4 tests produced a 0% mislabelling rate for non hate speech). However, they also produced at very high rate of mislabelling hate speech. For example, the GPT-2 generated data at 2000 rows SVM had about an 86% mislabelling rate. BERT saw a 45% mislabelling rate for the same training and test set. We decided to create a new test set that had an even 1:1 distribution of hate speech and non hate speech in order to see if the model was less effective on hate speech. However, we actually saw an overall improvement in the accuracy. Previously 86%, SVM had dropped to a 64% mislabelling rate. BERT had also improved, going from 45% to 27% for the same training and test data sets. The first test set was only selected from a section of the data. We did not take a sample. However for the second test, we took the rows more randomly. Because the training models over GPT-2 generated data actually improved when more hate speech was sampled into the test set, it gives us even more confidence that the GPT 2 data can be a reliable substitute for real data.

c. Comparison between SVM and BERT:

Looking at the baseline results, BERT outperforms SVM in all 3 categories of precision, recall, and F1 scores when they are trained and tested on the same datasets. The difference in effectiveness for raw processed data is about 20% to 25% across all experiments. The difference in effectiveness for GPT-2 processed data is about 15% to 20% across all experiments.

SVM:

- Lacks a well defined dictionary
- Relies on given data, taking the most frequently occurring tokens and uses that information to decide.
- Dataset of tweets used lacked clear patterns and contained multiple grammar/spelling errors.
- SVM tries to separate binary data. It seeks to maximize the distance of classes from the center. However, because 5000 features were used, it is likely that the data would be more clustered together and be unable to separate easily. In order to get around that, various kernels were used, but scores did not seem to beat BERT.

BERT:

- As a deep learning model, BERT is more sophisticated in it's approach.
- It never has a linear classification.
- Has its own dictionary that it is pre-trained on.

- Requires more time than SVM (SVM experiments took about 1-2 minutes, BERT took over an hour on the full original dataset).
- Included in the transformers hugging model, and can adapt to a optimizer. AdamW was used in this experiment.

d. The data shows a large increase in precision, recall, and f1 scores in BERT than SVM in all experiments. This could be because of the reasons mentioned above. The SVM seems to improve in precision, recall, and f1 as the amount of training data increases, but it seems that the baseline tests are generally better than the generated data tests. This could be because GPT-2 does not generate hate speech data that is as good/accurate as original, human created tweets. For BERT, an increase in scores is seen as the generated data increases, for all experiments but the one without any processing. This again is likely caused by poor GPT-2 text generation, especially because there would have been much noise in the tweets without any processing (both in the data that was used as input for GPT-2 generation, and the generated tweets). When comparing the SVM rbf (default), sigmoid, and polynomial kernels, the data suggests that rbf and sigmoid work better than the poly kernel. More research will need to be done into those functions to explain the differences in results. However, the polynomial seems to be unaffected by the amount of input data given.

e. Overall, the data seems to show that if GPT-2 inputs and text generated data were processed, it would help to train a model, BERT or SVM, to recognize hate speech. If given sufficient time, BERT will outperform SVM, and it would be the recommended tool for training a model to recognize hate speech.

f. If companies or organizations require the use of a hate speech classifier, we would recommend first seeing how accurate of a classifier they require and how much it would cost. It is much cheaper to generate your own input data using GPT-2 rather than scraping social media platforms for real data for training a model. If a lower accuracy is acceptable, GPT-2 generated text can be a good substitute for real data.

g. Because we see that GPT-2 data does not perform as well as real data, one could speculate that as the data set grows, the difference in accuracy may increase as well. We have determined that, in our experiments, GPT-2 data can perform similarly to real data for hate speech recognition. However, we would need to do more testing to find out how many rows of data models need in order to see a significant difference in accuracy. We have not trained the model on extremely large data sets (over 50,000 or 100,000 rows of training data), and we would have to complete more experiments in order to see if larger sets of GPT-2 data using processed inputs would create a more significant difference between the model trained on GPT-2 generated data and the model trained on real data. However, up to 2000 rows, we do not see much of an increase in the difference.

Task Table

Student	Summarized Tasks for All Sprints
Anjali	1. Steps for using original dataset as input to GPT2 2. Understanding and BERT model creation for using it on input dataset 3. Pre-processing steps, post-processing steps and additional noise removal steps 4. Generated 3 datasets for 500 records including 200 original records and 300 generated records. Dataset 1 - without preprocessing Dataset 2 - with preprocessing original data Dataset 3 - with preprocessing original data and post processing generated data 5. SVM and BERT analysis and experiments for 500 records using above datasets 6. Analysis of BERT performance for different epoch and batch size values
Elliot	1. Remove excess data columns 2. Researched how to use GPT 2 3. Create function to generate more training data from specific class using GPT 2 4. Run experiment for 1000 rows in BERT 5. Formatted and cleaned up BERT python file for readability 6. Helped analysis of BERT results
Akhilesh	1. Research and set up BERT for baseline + generated data experiments 2. Set up BERT hyperparameter 3. Contribute to generating data with GPT2 4. Perform most of BERT GPT2 experiments 5. Analyze mislabelling of GPT2 on BERT (using all training sizes)
Alexis	1. Research and setup SVM for baseline + generated data experiments 2. Contribute to pre/post-processing function 3. Generate data for experiments using GPT2 4. Perform all experiments 5. Experiment with SVM kernels 6. Analysis of performance of BERT vs SVM