Report on Tweets Classification

Nuraiym Ayatkyzy

November 2024

Abstract

Due to the nature of this research project, this report may include examples of hate speech and offensive language, which are used for research purposes only. The inclusion of such content is not intended to offend or harm anyone.

The growth of social media platforms led to harmful content threatening the inclusivity of online space. In fact, for a platform such as X (formerly Twitter), with approximately 430 million users worldwide, content moderation is more important than ever. This project aims to address this issue by developing and comparing three natural language processing (NLP) models to classify tweets as "hate speech," "offensive language," or "neither". In order to train models, a labeled dataset of real-life tweets is used, and multiple experiments are conducted to evaluate performance of each model. By offering an efficient solution for content moderation, this project attempts to contribute to fostering a safer online environment.

1 Introduction

The fast development in NLP over recent years has introduced a variety of research areas. Text classification that uses semantic analysis techniques is one of such areas. The emergence of hate speech, offensive statements, and, in general, harmful content within social media, have played a significant role in further advancing text classification techniques.

What role does X play in this discussion? If you consider the fact that X has about 430 million users globally, you cannot ignore its power in serving as a venue for harmful behavior [1]. In addition, according to The New York Times, hate speech on X has skyrocketed after Musk's acquisition: insults directed at Black Americans increased by about 202%, and slurs against gay men increased by 58% [2]. The increase in harmful content appears directly linked to the layoff of nearly 50% of X's content moderation team, reducing the platform's ability to control such behavior [3].

In general, the scale of content shared daily and complex nature of language makes hate speech detection a challenging problem. Social media platforms try to tackle this issue by implementing both automated and manual content moderation techniques. Nevertheless, the problem still persists because automated systems struggle with false positives and negatives, while manual review is resource-intensive. These challenges show the need for more advanced yet efficient content filtering systems.

1.1 Objectives and Scope

Hate speech is a form of language that targets person's or group's identity, such as religion, race, gender, etc. The goal of tweets classification is to develop a model that is able to accurately classify tweets as "hate speech," "offensive language," or "neither" by utilizing datasets available in Kaggle. Although, it is easy to differentiate between non-negative and negative tweets, the distinction between hate speech and offensive language is not always clear. The scope of the project extends beyond model development and evaluation, as it also explores the challenges associated with language biases and dataset limitations.

1.2 Literature Review

As it was mentioned earlier, the ease of information spread in the social media space led to text classification, in particular, hate speech detection, being a major task in the NLP area. That is why various models and technologies were developed in recent years.

In their survey paper, Schmidt and Wiegand [4] discuss how hate speech detection evolved over time. The biggest takeway from their work is that the basic techniques like bag-of-words or embeddings usually

provide decent results, arguing that character-level methods usually outperform token-level approaches. Besides that, they mention the usefulness of slur lists along with other features like meta-information in detecting hate speech.

Another work that explores the methods for hate speech detection was done by Dacon et al. [5]. In their research, it is revealed that in comparison with BERT surpassed HateBERT (a model pre-trained specifically for detecting hate speech), and all other models, achieving an F1-score of 0.82 across all labels. The similar result was reached by Malik et al. [6]. Researchers highlight that transformer-based models, such as BERT, ELECTRA, and Al-BERT outperformed other models. Opposite conclusion about BERT was done by Khan et al. [7]. The research points out that BERT could not differentiate between the offensive statement and neutral statement that contains offensive words. Additionally, data augmentation and ensemble strategies did not show much contribution in hate speech detection.

The work by Zhang et al. [8] is particularly relevant to this study because it analyzes tweets dataset. Researchers explore that using lexico-semantic features improved the effectiveness of simple LSTM models and SVM-based classifiers.

1.3 Hypothesis

Incorporating lexico-semantic features into neural networks like LSTMs or transformer-based models such as ELECTRA-Small can significantly enhance performance in hate speech classification. This will help to outperform simpler classifiers with basic feature engineering.

2 Data

The Hate Speech and Offensive Language (HSOL) dataset was developed using the hate speech lexicon from Hatebase.org. Using Twitter API, the total of 85.4 million tweets were collected. Consequently, a random sample of 25,000 tweets was selected for manual annotation into one of three categories: "hate speech," "offensive language," or "neither" [9]. However, as it can be seen from the table below, the data is imbalanced.

Label	Count
1	19,190
2	4,163
0	1,430

Table 1: Label distribution in the dataset.

3 Methodology

Since the aim of this project is to figure out most efficient model, the three different models are developed and compared in performance. Since the dataset is imbalanced, the model performance is first evaluated without SMOTE applied, and then compared with the performance when SMOTE is applied.

3.1 Models

3.1.1 Logistic Regression Classifier with TF-IDF

The text data is processed for classification by removing URLs, mentions, non-alphabetic characters, and stop words; converting text to lowercase; and lemmatizing words to their base form. The processed text is then converted into numerical representation using TF-IDF vectorization limited to 2000 features to balance complexity and performance. A logistic regression model is trained on the TF-IDF features and corresponding labels.

3.1.2 Hybrid LSTM Model

Almost the same preprocessing steps are applied to raw data along with tokenization limited to 10,000 words, and padding with maximum length 100. A sequential neural network model consists of embedding layer (learns word representations), an LSTM layer (captures sequential dependencies), a dense layer

(extracts features), a dropout layer (regularization), and the output layer with softmax activation. The model is compiled with the Adam optimizer, sparse categorical cross-entropy loss, and accuracy as the evaluation metric. A training is performed over 20 epochs, with a batch size of 32 and with early stopping based on accuracy.

3.1.3 Fine-Tuned ELECTRA-Small

The raw data is cleaned using the previously mentioned steps. Additionally, the dataset is tokenized with the ELECTRA tokenizer, with truncation and padding to a sequence length of 128 tokens and formatted as PyTorch tensors. A training is performed over 3 epochs with a batch size of 32, a learning rate of 2e-5, and early stopping is based on accuracy.

4 Numerical Experiments

4.1 Metrics

All models are evaluated on accuracy, f1 score, and ROC-AUC (One-vs-One). These metrics provide a detailed analysis of strengths and weaknesses of each model.

4.2 Evaluation

Metric	TF-IDF LR (No SMOTE)	TF-IDF LR (SMOTE)
Accuracy	0.8929	0.8433
F1	0.8769	0.8596
ROC AUC	0.9188	0.9176

Table 2: Model performance of TF-IDF LR with and without SMOTE on test data.

Metric	LSTM (No SMOTE)	LSTM (SMOTE)
Accuracy	0.8997	0.8257
F1	0.8865	0.8477
ROC AUC	0.9182	0.8763

Table 3: Model performance of LSTM with and without SMOTE on test data.

Metric	ELECTRA (No SMOTE)	ELECTRA (SMOTE)
Accuracy	0.9056	0.8182
F1	0.8796	0.8514
ROC AUC	0.9473	0.9438

Table 4: Model performance of ELECTRA with and without SMOTE on test data.

4.3 Test cases

The generalization capability of models are evaluated based on three test cases for each class. For the "hate speech" class, the following texts will be tested: *"I think Mexicans should be deported," "I hate Asians," "Women can't be good politicians."* For the "offensive language" class, the test cases include: *"You are fat and ugly, go lose weight," "Stupid bitch," "You are a family disappointment."* For the "neither" class, the sentences are: *"Hello," "You are so pretty," "Today is a good day."*

Additionally, to see if models understand context, a special test cases include sentences with negative words but positive or neutral contexts: *"I do not hate Asians," "Mexicans should not be deported," "Stop, you are not ugly but so pretty."* This contextual test ensures that the models do not rely on the presence of specific words only and can understand the overall sentiment.

Test Case	TF-IDF LR (No SMOTE)	TF-IDF LR (SMOTE)
I think Mexicans should be deported	Offensive Language	Hate Speech
I hate Asians	Neither	Hate Speech
Women can't be good politicians	Offensive Language	Offensive Language
You are fat and ugly, go lose weight	Offensive Language	Hate Speech
Stupid bitch	Offensive Language	Offensive Language
You're worthless	Offensive Language	Neither
Hello	Offensive Language	Neither
You are so pretty	Offensive Language	Neither
Today is a good day	Neither	Neither
I do not hate Asians	Neither	Hate Speech
Mexicans should not be deported	Offensive Language	Hate Speech
Stop, you are not ugly but so pretty	Offensive Language	Hate Speech

Table 5: Performance of logistic regression classifier with TF-IDF on test cases.

Test Case	LSTM No SMOTE	LSTM SMOTE
I think Mexicans should be deported	Hate Speech	Hate Speech
I hate Asians	Neither	Neither
Women can't be good politicians	Neither	Neither
You are fat and ugly, go lose weight	Neither	Neither
Stupid bitch	Offensive Language	Offensive Language
You're worthless	Neither	Hate Speech
Hello	Neither	Neither
You are so pretty	Neither	Hate Speech
Today is a good day	Neither	Hate Speech
I do not hate Asians	Neither	Neither
Mexicans should not be deported	Hate Speech	Hate Speech
Stop, you are not ugly but so pretty	Neither	Hate Speech

Table 6: Performance of LSTM model on test cases.

Test Case	ELECTRA No SMOTE	ELECTRA SMOTE
I think Mexicans should be deported	Neither	Neither
I hate Asians	Neither	Neither
Women can't be good politicians	Neither	Neither
You are fat and ugly, go lose weight	Neither	Neither
Stupid bitch	Offensive Language	Offensive Language
You're worthless	Offensive Language	Hate Speech
Hello	Neither	Neither
You are so pretty	Neither	Neither
Today is a good day	Neither	Neither
I do not hate Asians	Neither	Neither
Mexicans should not be deported	Hate Speech	Neither
Stop, you are not ugly but so pretty	Neither	Neither

Table 7: Performance of ELECTRA model on test cases.

4.4 Discussion

Logistic regression with TF-IDF (without SMOTE) performed poorly in test cases, correctly classifying 5 of 12 test cases. This poor performance is mainly due to the imbalanced dataset, with majority of instances being from "offensive language" class. This explains why the model classified 9 out of 12 test cases as "offensive language," and none as "hate speech." Additionally, TF-IDF calculates the frequency of each word in a specific class and cannot capture the context, which also explains such performance.

With SMOTE applied, the model improved for some categories but still struggled in others. It

correctly predicted 2 out of 3 hate speech examples, 1 out of 3 offensive language examples, 3 out of 3 neutral examples, and 0 out of 3 contextual examples. At a closer look, the working principle of TF-IDF is evident: thanks to "clues" ("Mexicans" and "hate"), it correctly classifies hate speech examples. At the same time, the model was unable to correctly classify contextual examples since they also contain words that are frequent in hate speech examples.

When it comes to LSTM with no SMOTE, it correctly classified 7 out of 12 test cases. However, it cannot account for the robustness of the models. In particular, in *"I think Mexicans should be deported"* and *"Mexicans should not be deported"* cases, the model classified both as "hate speech" due to the presence of word "Mexicans", which was possibly encountered during training. Similarly, in *"I hate Asians"* and *"I do not hate Asians"* cases, the model classified both as "neither," which accounts for the model's inability to generalize and limitations of the dataset. Implementing SMOTE did not help either and the same issues persisted.

The ELECTRA-Small model without SMOTE correctly classified 7 out of 12 test cases, with the majority of all classifications being from underrepresented "neither" class, which might point to unnecessity in SMOTE. The most interesting misclassifications are *"I think Mexicans should be deported"* and *"Mexicans should not be deported"*. One of the possible reasons for that to happen is model's focus on certain tokens more than the overall sentence structure. Besides that, the phrase "I think" might reduce the hateful sentiment of the whole sentence. When SMOTE was applied, it still correctly classified 7 of 12 test cases. Another observation is that out of 3 test cases for "offensive language" class, only *"Stupid bitch"* was correctly classified both with and without SMOTE, probably because of the presence of explicitly offensive word "bitch."

Investigation of validation metrics alone shows that, overall, ELECTRA without SMOTE performed the best. Nevertheless, ELECTRA's classification accuracy in test cases was average, possibly because of the difference in tweets' and test cases' style, and dataset limitations.

5 Conclusion

The limited size of the dataset led to models' inability to generalize. Simple classifier like Logistic Regression with TF-IDF struggled in handling minority classes and understanding context. While SMOTE slightly improved predictions, TF-IDF's inability to understand semantics was evident. The LSTM and ELECTRA-Small also incorrectly classified context-dependent examples because of their reliance on token frequency and inability to process sentence-level semantics. Despite that, in general, the hypothesis was validated because more advanced models outperformed simple classifier. The findings indicate that further integration of advanced lexico-semantic features and larger dataset will improve models' performance in hate speech detection.

References

[1] "X (formerly Twitter) - statistics & facts." *Statista.* Available: https://www.statista.com/topics/737/twitter/topicOverview (accessed Nov. 21, 2024).

[2] S. Frenkel & K. Conger. "Hate speech's rise on Twitter is unprecedented, researchers find." *The New York Times.* Available: https://www.nytimes.com/2022/12/02/technology/twitter-hate-speech.html

[3] "Musk fires outsourced content moderators who track abuse on Twitter." *CBS News.* Available: https://www.cbsnews.com/news/elon-musk-twitter-layoffs-outsourced-content-moderators/

[4] A. Schmidt and M. Wiegand. "A survey on hate speech detection using natural language processing," in *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media,* 2017, pp. 1–10.

[5] J. Dacon, H. Shomer, S. Crum-Dacon, and J. Tang. "Detecting harmful online conversational content towards LGBTQIA+ individuals," 2022.

[6] J. S. Malik, G. Pang, and A. van den Hengel. "Deep learning for hate speech detection: A comparative study," 2022.

[7] Y. Khan, W. Ma, and S. Vosoughi. "Lone Pine at SemEval-2021 Task 5: Fine-grained detection of hate speech using BERToxic," 2021.

[8] H. Zhang, M. Wojatzki, T. Horsmann, and T. Zesch, "LTL. UNI-DUE at SemEval-2019 Task 5: Simple but effective lexico-semantic features for detecting hate speech in Twitter," in *Proceedings of the 13th International Workshop on Semantic Evaluation,* 2019, pp. 441–446.

[9] "Hate Speech and Offensive Language." *Papers With Code.* Available: https://paperswithcode.com/dataset/hate-speech-and-offensive-language