LING490: Hate Speech Detection System

Aditi Khanna
University of Illinois,
Urbana Champaign
aditik4@illinois.edu

Ariane Taraki
University of Illinois,
Urbana Champaign
atarak2@illinois.edu

Erik Ly
University of Illinois,
Urbana Champaign
erikly2@illinois.edu

Abstract

Billions of people use social media everyday for a variety of reasons on a variety of different platforms: Facebook, Instagram, Twitter. But with the immense usage of social media comes an equally immense amount of hate speech. Having a platform for individuals to speak freely comes with its costs, namely in an upsurge of racist and sexist comments. And while the number of social media users only continues to grow exponentially, companies do not exactly have enough manpower to shut down all hate speech themselves. Therefore, there is a need to be able to automatically detect these hateful/negative messages before they are sent out, to clear the social media atmosphere of hurtful speech.

In this paper, we discuss our findings from our explorations in detecting hate speech in on the social media platform Twitter. We categorize our hate speech into three categories, "racist", "sexism", and "none". We aim to highlight the process of how we were able to detect these categories utilizing data from twitter users. We then provide an analysis of the three different models utilized on our data to see how well our models performed and the accuracy of our system. Naive Bayes, Decision Tree, and Linear Regression approaches were all applied to form our system. Error analysis on the results and discussion of these results conclude to prove that the Decision Tree model performs the best with the quantity and type of data parsed in this problem.

Results and data used can be found on the project GitHub repository.[1]

1 Introduction

It is widely understood that much of hate speech is ill-defined in the linguistics community. Social and cultural definitions on what it means for language to be hate speech complicate attempts for computational linguistics to attack the issue of parsing such data. While there exists a baseline for what words or phrases are categorized as hateful, much of the negative comments are based on context and are often subjective.

As regular users of social media ourselves, we have all seen the amount of hate comments and sexist/racist speech on platforms such as Twitter. Social media platforms oftentimes attempt to mitigate these issues by placing moderators or bots in place to capture hateful speech – but the definition of what this speech is, and how to better detect said speech is complicated.

Many people argue that online platforms should not be allowed to regulate "free speech" as it is seen in the American definition, while others argue that hate speech has consequences that can be disastrous (especially for generations growing up with the internet) and therefore should not exist on the internet. Through pop culture and our own

[1] https://github.com/erikly2/LING490-Hate-Speech-Detection

lived experiences, we can see just how disastrous the effects of hate speech can have on individuals – as words carry more meaning than meets the eye.

As the fields of Natural Language Processing, as well as Machine Learning are increasingly expanding in application, we see more possibility of technology being able to tackle hate speech accurately and successfully. Our system provides a benchmark towards this by creating an avenue for technology to filter out these hateful comments.

Therefore, the work we present here – creating a system to detect hate speech in tweets – can be used as a starting place for linguists to detect and flag racist and sexist speech on various platforms (i.e., Twitter in this case). By doing this, one would be able to detect when a social media environment is fostering hate speech, thus providing an opportunity to combat the speech in general to help create a more inclusive community. Additionally, our error analysis can be useful for other researchers in the field for determining how different ML models respond when used in a system of this context.

2 Problem Definition

The problem our group has attempted to solve is creating a system that can detect hate speech (sexism and racism) in text (in this case, specifically tweets).

Using an already existing annotated dataset as the foundation of our work, we created a new dataset of tweets (obtained from the Twitter API), and then built our system by training and testing multiple ML models and presenting an error analysis on the respective results.

The input to our system is a list of tweet IDs annotated with their respective labels (sexism, racism, or none). The output to our system is a group of results given by each ML model– Naive Bayes, Linear Regression, and Decision Tree. We produce values including r2, precision, recall, and accuracy which can be compared amongst models and provide insight on the validity of our system.

The final result of our system comes in the form of our error analysis and conclusion, determining how well our models detected hate speech when trained on the tweet data.

3 Previous Work

Although there is not extensive prior research in the field of hate speech detection, we gathered and leveraged data from research contained in various works related to this field.

The main focus for detecting hate speech issues revolves around work conducted in Detecting Offensive Language in Social Media to Protect Adolescent Online Safety (Ying Chen et al., 2012). As mentioned in the paper, although social media has allowed adolescents to communicate with a wider audience, it has also allowed adolescents to be targeted with harmful content. Content discussed in the research here is important towards understanding the impacts that hateful language has on people – especially young adults and children. As more and more generations are growing up online with unlimited access to social media, concerns towards the effects of language online on adolescent growth also increases. Taking these concerns into consideration, being able to implement functioning systems to detect and eliminate hate speech online would be very beneficial to all involved.

Much of the research done in our system was conducted off the foundation of Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter (Zeerak Waseem and Dirk Hovy, 2016). The subsequent dataset provided from this research was created from a model that served to detect racism and sexism within tweets submitted by users from the Twitter API. This dataset consists of 16k+ different tweet IDs, filtered, and annotated with its respective hate speech classification. These tweets were sent and annotated by users with the following breakdown: 3,383 for sexist content, 1,972 for racist content, 11,559 for neither content.

In addition, we found similar issues to that of Hate speech detection: Challenges and solutions (MacAvaney S et al., 2019). Researchers here found that the definition of hate speech changed from person to person along with how blurry the lines between hate speech and free speech is. To fix

this issue, they came up with a clear definition of hate speech categorized into four parts. They would then fact check the statement or check if it supports certain groups to determine if it was hate speech or free speech. Although we did not use this exact methodology, we utilized the approach these researchers were going for to create a model that closely resembles their approach.

After we finished our models, we used Racist detection in Dutch social media posts (Stephan Tulkens et al., 2015) to compare our precision, recall, and f1 scores. We did this to verify the accuracy of our model and how it compared to other researchers in the area. Overall we found that higher scores on our decision tree compared to the study.

4 Approach

Our approach is sectioned as follows: Parsing of Data, Machine Learning Modeling, Error Analysis.

4.1 Parsing of Data

We first used the tweet data set[2] from Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter (Zeerak Waseem and Dirk Hovy, 2016) as the foundation for our work. We gathered these tweets by parsing the Twitter IDs in question from the Twitter API, storing them for usage alongside their annotated categorization ("sexism", "racism" or "none").

Since we were taking the already annotated dataset from previous research, we thought it best to not do any further preprocessing. If the tweet was unable to be found or we were not authorized to view the tweet, we replaced that tweet's content with "no data" to keep track of what data was unavailable.

4.2 Machine Learning Modeling

The parsed tweet data was then used for training and testing on our Machine Learning models. We chose three traditional models (Naive Bayes, Decision Tree, Linear Regression) to create our system with.

[2] https://github.com/zeeraktalat/hatespeech/blob/master/NAACL_SRW_2016.csv

R2 Training Score: 0.988
R2 Testing Score: 0.028

	precision	recall	f1-score	support
none	0.68	0.08	0.14	239
racism	0.00	0.75	0.01	4
sexism	0.00	0.00	0.00	538
accuracy			0.03	781
macro avg	0.23	0.28	0.05	781
weighted avg	0.21	0.03	0.04	781

Figure 1: Results of the Naïve Bayes model performance

We chose these models, as we are historically more familiar with them through our coursework and had more confidence in being able to better understand and come to conclusions based on these more traditional models in contrast to other more difficult deep-learning models. Additionally, by using multiple Machine Learning models and comparing them to each other, we were more accurately able to come to conclusions as to how and why our models perform differently.

The Naïve Bayes model (Gaussian) and Decision Tree model were completed similarly, using tweet content itself as X data, and classification labels ("racism", "sexism", "none") and Y data for training and testing. String content was converted into float values for modeling, and sklearn library attributes were used to create the models and return metric results on the data.

For both the Naïve Bayes and Decision Tree models, we gathered R2 Testing and Training scores, as well as an overall classification report on precision, recall, f1-score and support.

Due to the Linear Regression model needing numbered values for computation, the data we

R2 Training Score: 0.988
R2 Testing Score: 0.703

	precision	recall	f1-score	support
none	0.68	0.08	0.14	239
racism	0.00	0.00	0.00	4
sexism	0.70	0.99	0.82	538
accuracy			0.70	781
macro avg	0.46	0.35	0.32	781
weighted avg	0.69	0.70	0.61	781

Figure 2: Results of the Decision Tree model performance

R2 Training Score: 0.024
R2 Testing Score: 0.021

Figure 3: Results of the Linear Regression model performance

parsed had to be converted to integers. To do so, values that were not racist or sexist had a score of zero, where if a tweet was racist, it would get a score of 0.8 and sexist would receive a value of 0.6. We then had to convert each tweet to a number so the model could complete its comparison. Therefore, we went through each tweet and counted how many keywords were contained in the tweet and used that value as a representation of the tweet. From there, the R2 score was computed for analysis.

4.3 Error Analysis

For our error analysis, we compared the resulting performance of our models, as well as why we believe the performance differences existed or did not exist.

This section was very important for the impact of our system, as we believe it can serve to be a foundation of understanding for researchers in this field as to how different Machine Learning models react to hate speech data in testing and training.

5 Results

The following are the performance results of the models in our system.

5.1 Error Analysis

As mentioned previously, our system included three models: Naive Bayes, Decision Tree, and Linear Regression. With the first two models, we have the R2 Testing, Training results, and a classification table of precision, recall, f1, support (see Figures 1 and 2). For the Linear Regression model, we could only output our R2 Testing and Training results, as other metrics are not available for a quantified model such as this one (see Figure 3).

We used the text in the tweets themselves, as well as their classification of “racism”, “sexism”, or “none” to train and test our models. Because of issues and limitations with access to data, we had very little data on “racism” to work with, which explains why our resulting metrics scored so low on that classification.

The model that performed best in almost every metric was the Decision Tree model. Not only did the Decision Tree model have the highest R2 scores, which represents correlation of data between X and Y, it also performed best in most precision, recall, and f1 metrics results. One reason we believe this is so, is because Decision Tree models typically obtain higher accuracies with smaller data needed. This is because it is a deterministic and discriminative model and is somewhat easier to work with on smaller scales (datasets). The Naive Bayes model is non-deterministic and generative, and additionally more effective on larger amounts of data (requiring more data to gain better accuracy).

In our case, because we did not have a huge amount of data to work with (as we had to parse through tweets and gather them from the Twitter API, often with tweets not being accessible and disregarded from the dataset), the Naive Bayes model did not perform as well as the Decision Tree one did. If we had had more data for the Naive Bayes model to work with, we believe it might have outperformed the Decision Tree one.

With regards to the Linear Regression model, it is usually accurate as long as metric data can be calculated with a formula, however it requires the quantification of data. This is why it did not perform well in our case -- as the data that was being parsed here was more content heavy, it was harder to quantify it into X and Y values for model testing.

6 Discussion and Conclusions

We learned a lot from this project, and all had a pleasant experience working with the data and models described above.

The From this project we have learned about how to work with the Twitter API to access data. This proved challenging because much of the information was privatized and unavailable for us to readily use.

Furthermore, we each were able to explore the implementation of several machine learning models and how to interpret those results.

There are several possibilities for improvements on our approach. For instance, we only used the text of the tweet as features for our model. However, we could add more features such as the use of emojis (quantifying the existence of emojis in tweets or not), and the tweet author’s information (such as location and age) if given, to

analyze demographic information and how that may play a role.

Overall, we enjoyed this project and how our team was able to work together to parse, train, and test twitter hate speech data on our models.

Acknowledgments

Special thanks to Professor Roxana Girju, our LING 490 professor who gave us the tools to complete this project.

References

MacAvaney S, Yao H-R, Yang E, Russell K, Goharian N, Frieder O (2019) Hate speech detection: Challenges and solutions. PLoS ONE 14(8): e0221152. https://doi.org/10.1371/journal.pone.0221152

Stephan Tulkens, Lisa Hilte, Elise Lodewyckx, Ben ´ Verhoeven, and Walter Daelemans. 2015. Detecting racism in dutch social media posts, 2015/12/18.

Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. 2012a. Detecting offensive language in social media to protect adolescent online safety. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Conference on Social Computing (SocialCom),* pages 71–80. IEEE, September

Zeerak Waseem and Dirk Hovy. 2016. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California. Association for Computational Linguistics.