Hate Speech Classification

Wen Zhang(Eva Wu, Nikky Yu)

2020/12/3

Disclaimer: the dataset for this research contains text that may be considered profane, vulgar, or offensive.

Introduction

As the internet has been utilized widely in recent years, people tend to favor talking and discussing treading events and topics online. As a result, commenting on issues and circumstances following the original posts has got more popular than before. However, while commenting on various subjects, when people are starting to get more emotional and intense, they tend to express their emotions through the word's utilization. Thus, there appears a large number of inappropriate words. As a result, discussing things that people care are about is getting more and more difficult. People face the threat of abuse and harassment when expressing their own ideas. The toxic comments negatively affect the cyber environment and negatively affect teenagers who also have access to these comments. Moreover, people might tend to not express themselves and close the comments option. The platforms then struggle to facilitate the conversations, which makes number of communities decide to limit and shut down the comments from user to avoid the negative impacts. Therefore, our goal is to locate the specific toxic identifier and utilize the identifiers to create a model that could detect the individual sub-categories(toxic, severe toxic, obscene, threat, insult, identity hate) those comments are.

Data Cleaning and EDA

The data we utilized is from the Conversation AI team. The research institute is working on generating tools to improve the online conversation. In our project, we are only focusing on negative online behaviors. The data we are working on is provided by Conversation AI team. Conversation AI team is a research initiative that is founded by Jigsaw and Google. Negative online behavior is one of their studies. The data contains large number of Wikipedia comments, and was labeled by human raters for toxicity, including toxic, severe_toxic, obscene, threat, insult and identity_hate. There are 159571 rows in the data. Each row contains id, text of the comments and 6 categorical variable which are sub-categories of toxicity including toxic, severe_toxic, obscene, threat, insult and identity_hate. The 6 sub-categories are created by human. Comments may fall into more than one category. For instance, some comments might be labeled as both toxic and identity hate. Below is a sample comments before data cleaning.

[1] "COCKSUCKER BEFORE YOU PISS AROUND ON MY WORK"

This comment is a typical negative comment that are being labeled as toxic, severe_toxic, obscene, and insult. Here is another example of comments before data cleaning.

[1] "Sorry if the word 'nonsense' was offensive to you. Anyway, I'm not intending to write anything

The second comment are example of non-toxic comment.

Data Cleaning

We first load the training data. We will do the data cleaning regarding to the text of the comment. We convert text data into data frame and remove the punctuation and digit exist in the comments that will obscure our data analysis. Then we utilized tibble() to convert text data into data frame. After that we eliminate the stop word that will also obscure our analysis. The stop word included “a”, “a’s”, “able”, “about”. They are all common words that show up frequently in the sentence but will obscure our analysis. The complete list of stop words can be access by data(stop_words). In order to implement analysis, we need to have a one-table-per-row table. We then tokenize the comments and reconstruct it into a one-token-per-row data using unnest_tokens. For instance, is a sample comments is “I eat 10 apples today”, after our data cleaning, the sentence will become “i”, “eat”, “apples”, and “today”.

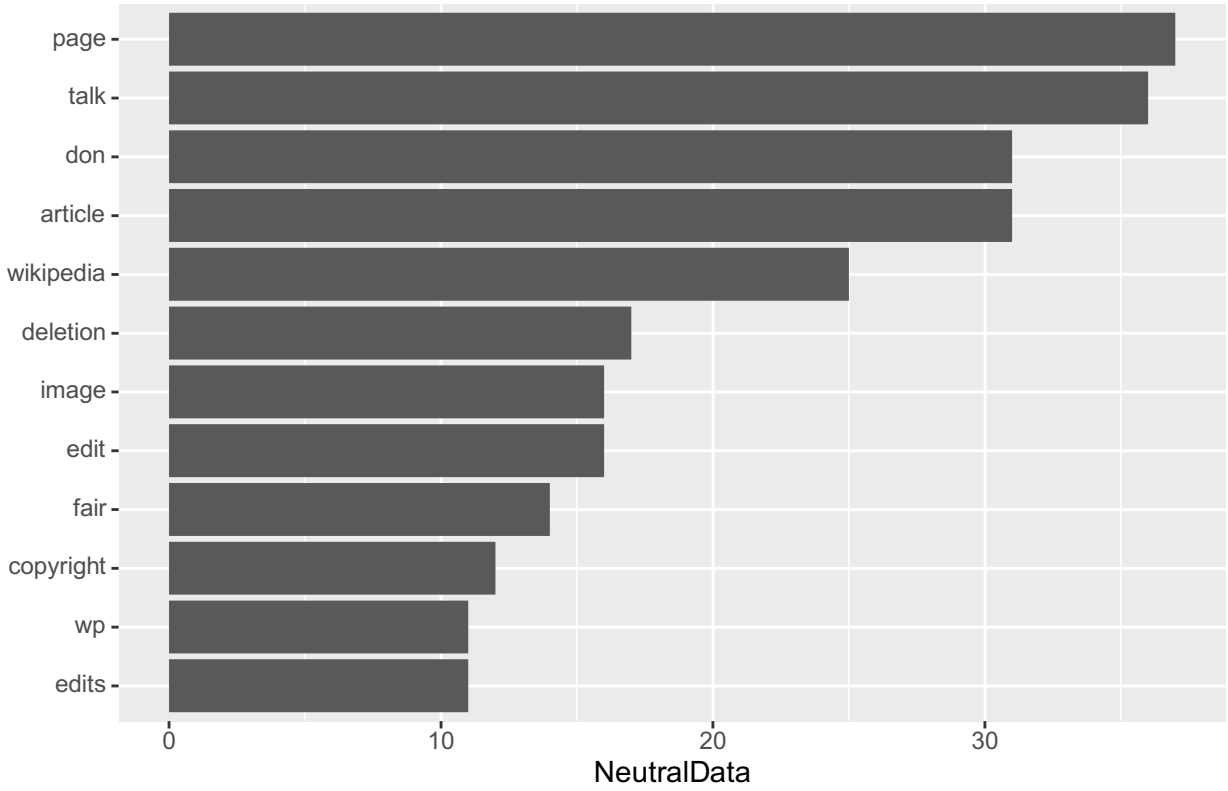
EDA

After the data cleaning, we are now going to closely observe the words and figure out the insights of the words. By doing that, we are able to find out identifiers that could be included in our models later.

We create two different sub-categories, “Nasty” and “Neutral”. “Nasty” represents that the comment falls into at least one of the categories of toxic, severe toxic, obscene, threat, insult, or identity hate. “Neutral” represents that the comment_text does not fall into any of the category including toxic, severe toxic, obscene, threat, insult, and identity hate. There are 16225 nasty comments and 143346 neutralcomments.

We then create visualization to observe the common words about all observations to gain the knowledge about the overall word frequency of the observations.

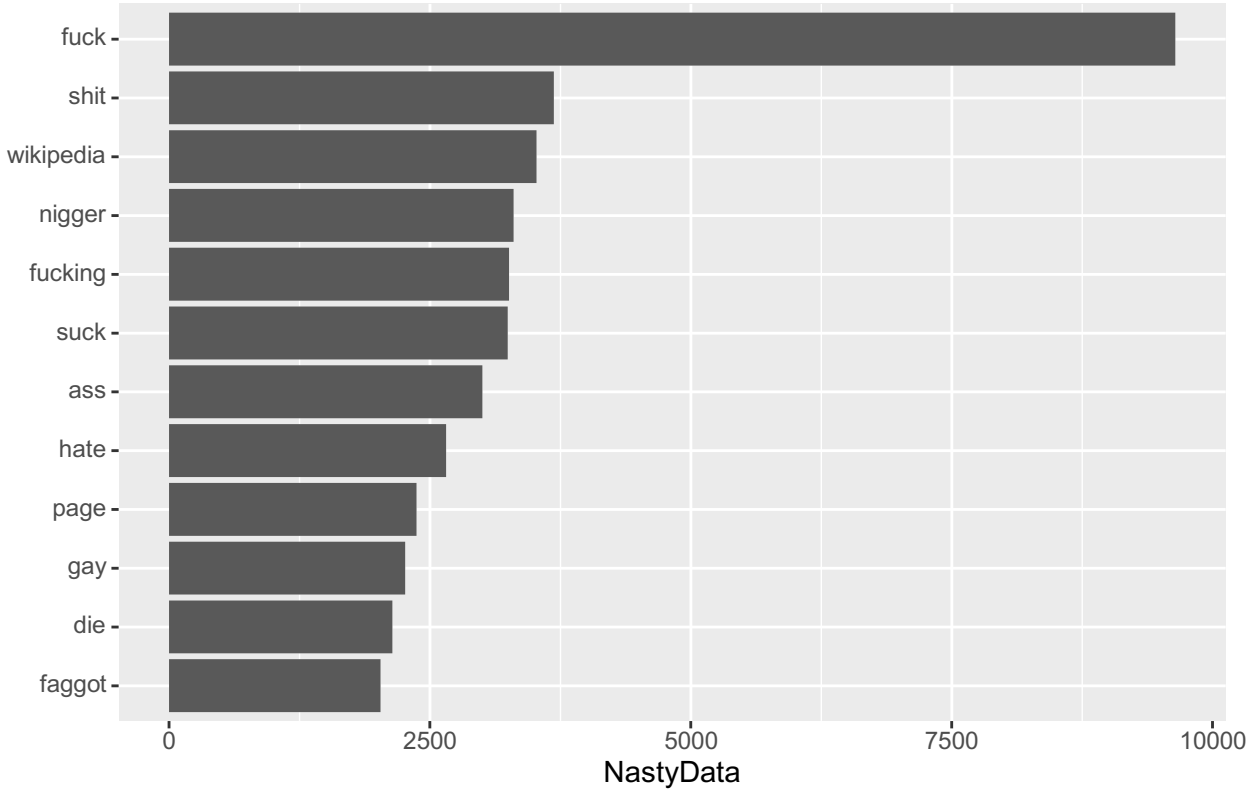
Figure 1.1



The top 10 words are page, talk, don, article, wikipedia, deletion, image, edit, fair, copyright. Since our goal is to come up with a model that could detect the sub-categories a certain comment belongs to, it makes

logically sense to more closely observe the negative comments. We then make a visualization that only focus on nasty comments.

Figure 1.2



The top 10 words are f-k, s-t, wikipedia, n-r, fu-k, su-k, a-s, h-e, p-e, and g-y.We can observe from both visulizaiton of nasty and neutral comments that the top words are different, but there are some overlapping, for instance "wikipedia" and "page". Also, neutral data visualization mostly captures neutral words while in nasty data visualization, most of the words are profane, vulgar, oroffensive.

We then closely looked at 6 sub categories including toxic, severe toxic, obscene, threat, insult, or identity hate to gain insights about words in individual categories.We created visualizations for the top frequent words for each sub-category separately.

Figure 1.3

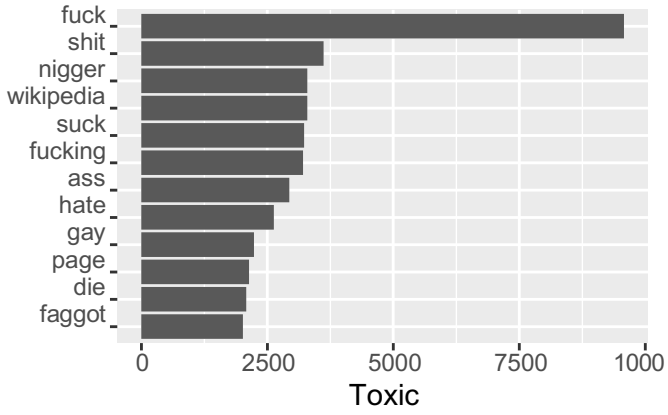


Figure 1.4

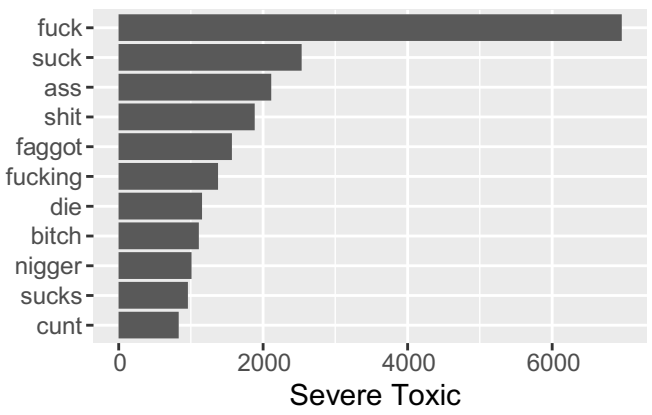


Figure 1.5

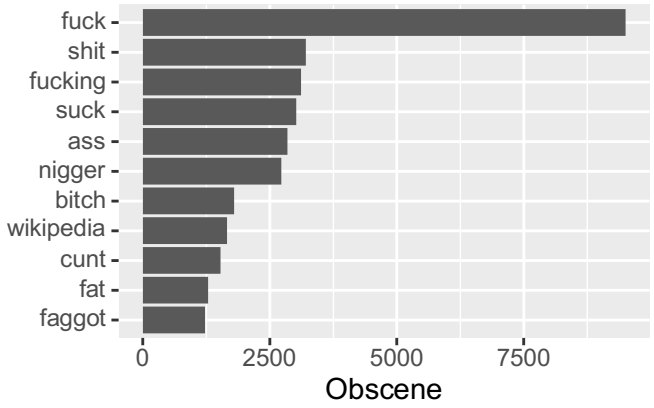


Figure 1.6

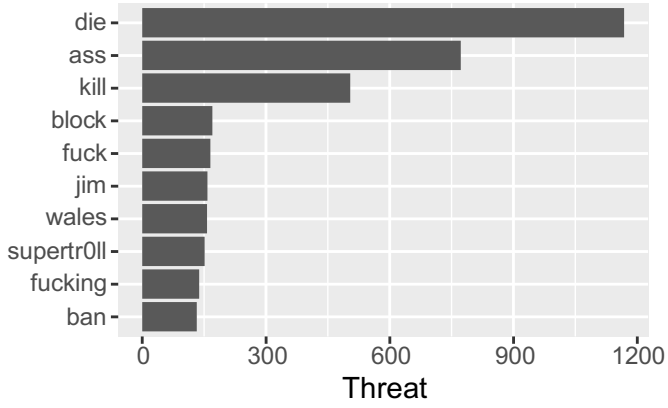


Figure 1.7

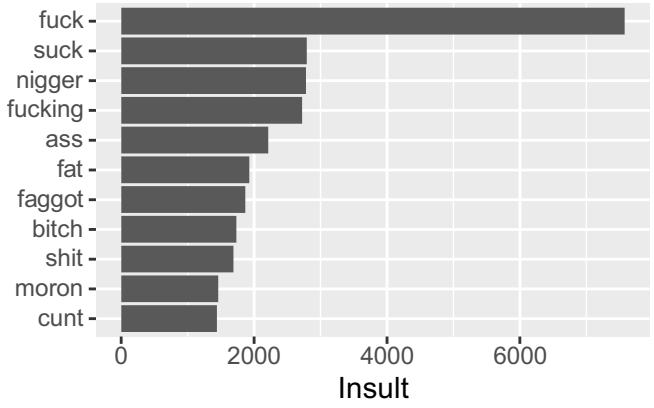
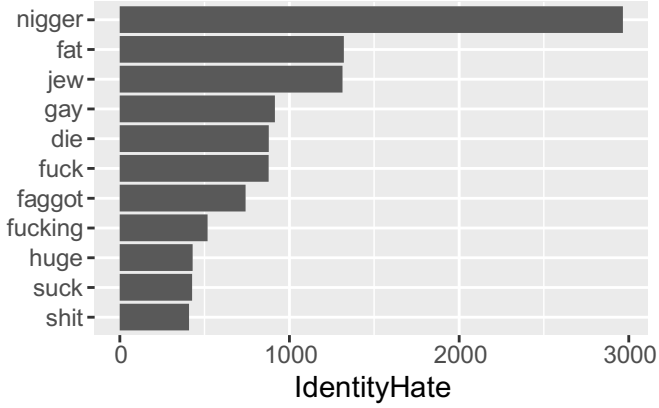


Figure 1.8



We found out the top 10 words for each category. Figure 1.3 is top word frequency for toxic comments. There are 15294 observations being labeled as toxic. The top 10 frequently appeared word are f-k, s-t, n-r, wikipedia,su-k, f-g, a-s, h-e, g-y, and page. Figure 1.4 is top word frequency for severe toxic comments. There are 1595 observations being labeled as severe toxic. The top 10 frequently appeared word are f-k, su-k, a-s, s-t,f-t, f-g, d-e, b-h, n-r, and su-s.Figure 1.5 is top word frequency for obscene comments. There are 8449 observations being labeled as obscene. The top 10 frequently appeared word are f-k, s-t, f-g, su-k, a-s, n-r, b-h, wikiedia, c-t, and f-t. Figure 1.6 is top word frequency for threat comments. There are 478 observations being labeled as threat. The top 10 frequently appeared word are d-e, a-s, k-l, b-k, f-k, j-m, w-s, s-l, f-g, and b-n. Figure 1.7 is top word frequency for insult comments. There are 7877 observations being labeled as insult. The top 10 frequently appeared word are f-k, su-k, n-r,f-g, a-s, f-t, fa-t, b-h, s-t, m-n, c-t. Figure 1.8 is top word frequency for identity hate comments. There are 7877 observations being labeled as identity hate. The top 10 frequently appeared word are n-r, f-t, j-w,g-y,d-e, f-k, fa-t, f-g, h-g, and s-k.

We also checked the overlap of each two categories. The result suggested that toxic, obscene and insult tend to have high correlation with each other. There are 48.85% of comments that are categorized as toxic and obscene, and 45.26% of comments that are categorized as toxic and insult.

To further compare the overlap between each category, we created 15 wordclouds. The flowing visualizations are word clouds for toxic and severe toxic(figure 2.1), toxic and obscene(figure 2.2), toxic and threat(figure 2.3), toxic and insult(figure 2.4), toxic and identity hate(figure 2.5), severe toxic and obscene(figure 2.6), severe toxic and threat(figure 2.7), severe toxic and insult(figure 2.8), severe toxic and identity hate(figure 2.9), obscene and threat(figure 2.10), obscene and insult(figure 2.11), obscene and identity hate(figure 2.12), threat and insult(figure 2.13), threat and identity hate(figure 2.14), and insult and identity hate(figure 2.15).

Figure 2.1

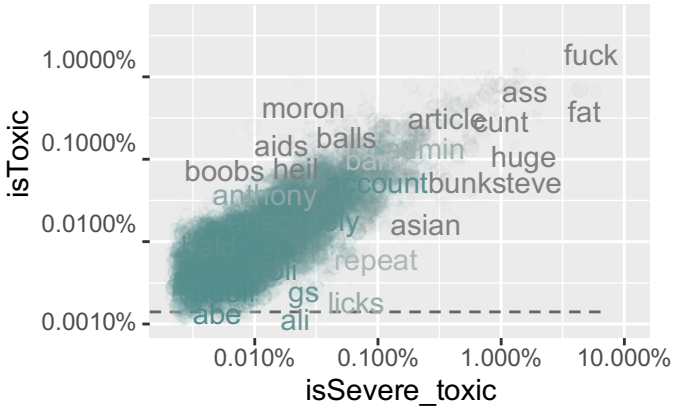


Figure 2.2

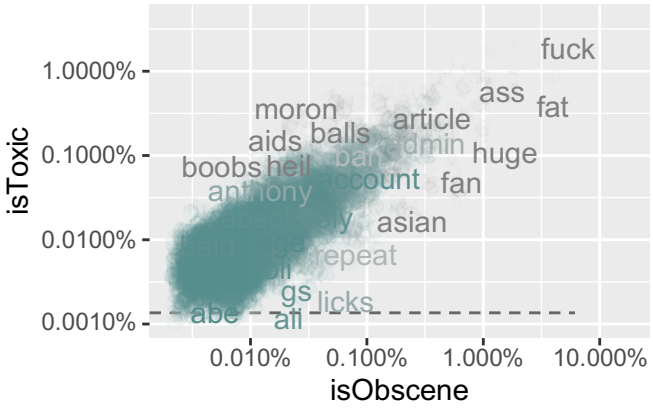


Figure 2.3

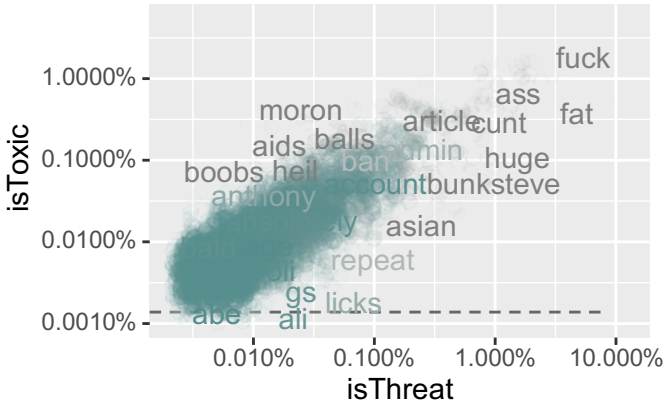


Figure 2.4

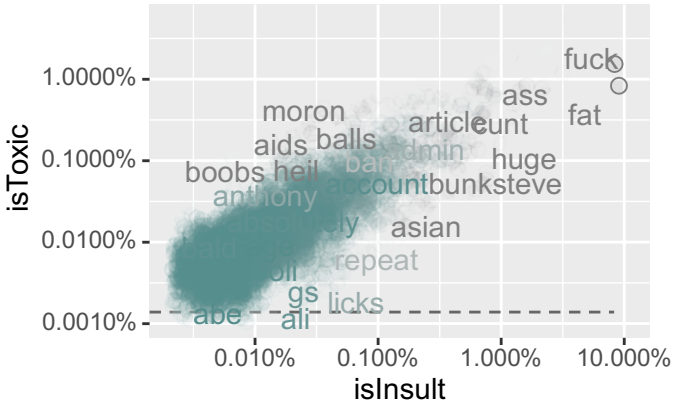


Figure 2.5

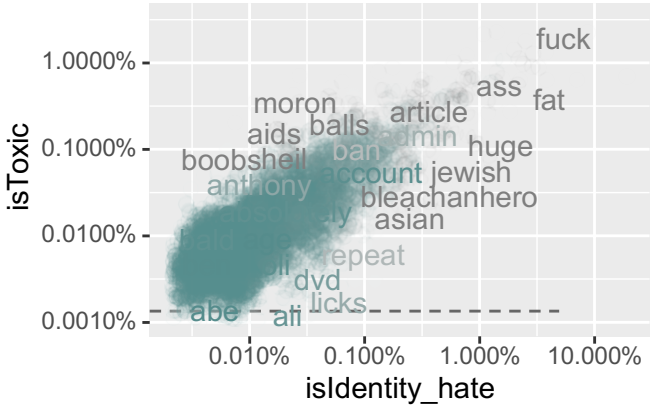


Figure 2.6

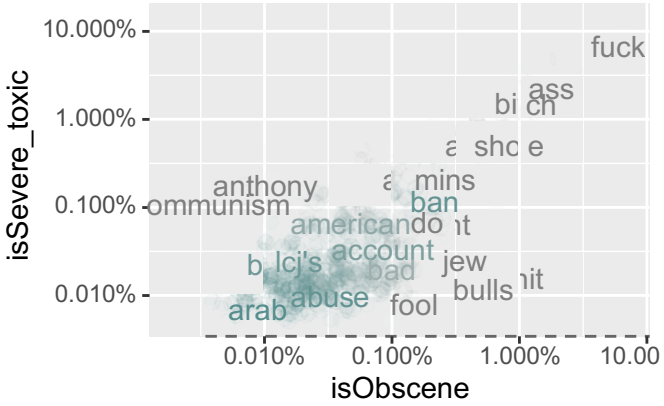


Figure 2.7

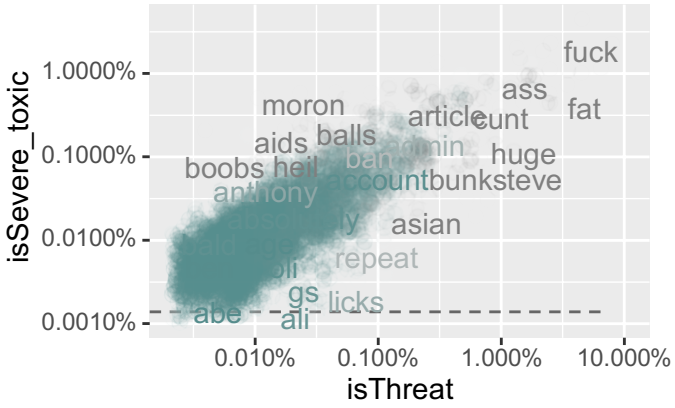


Figure 2.8

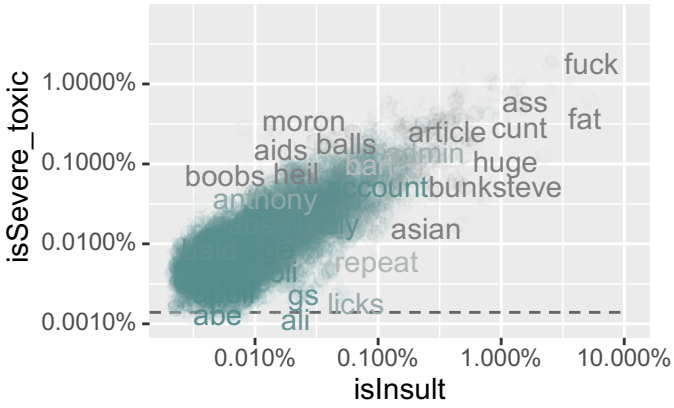


Figure 2.9

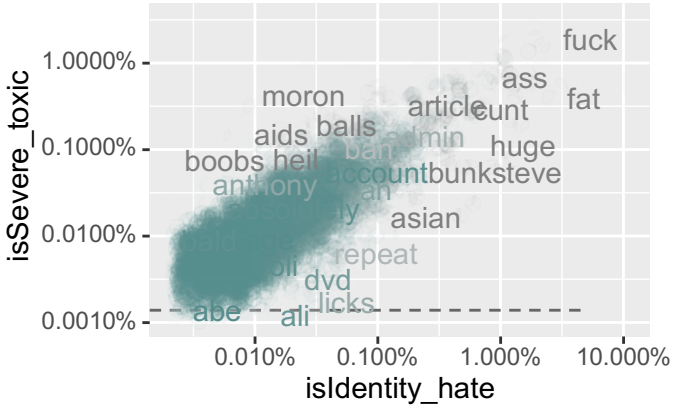


Figure 2.10

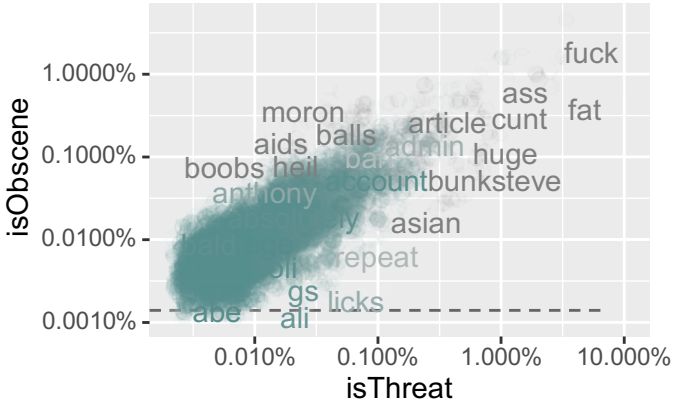


Figure 2.11

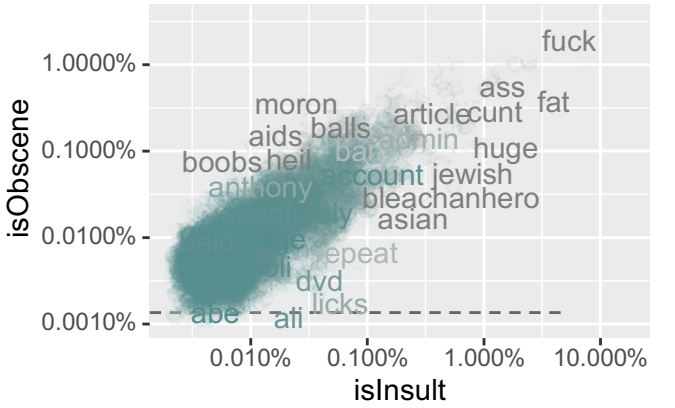


Figure 2.12

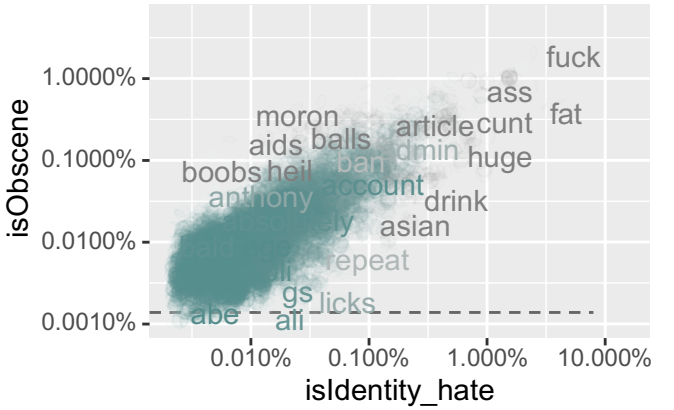


Figure 2.13

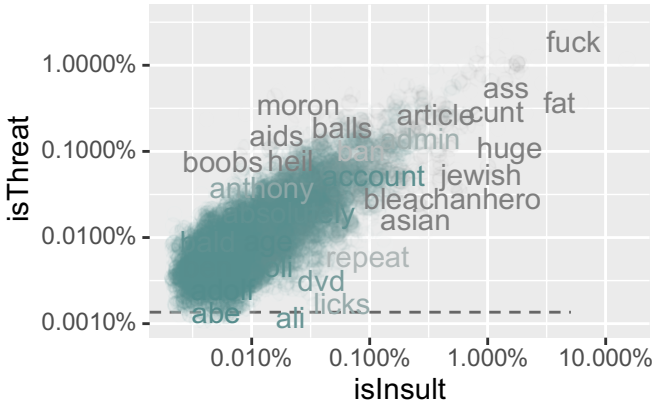


Figure 2.14

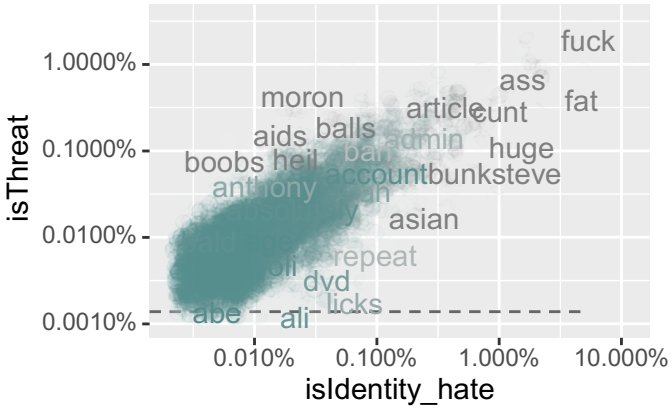
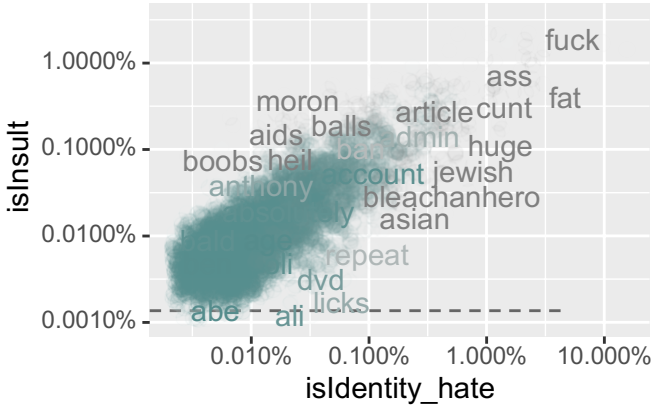


Figure 2.15



In Figure 2.1, the word cloud for toxic and severe toxic, we can observe the word frequency for both sub-categories of toxic and severe_toxic. The y variable in the graph represent the frequency of word for toxic comments and the x-axis represent the frequency of word for severe_toxic comments. Every data point in the graph represent a individual words that appear both in toxic and severe_toxic comments. The words that clustered on and near the line in the graph is those words that have similar frequency of occurring in both toxic and severe toxic comments. The words that shows up in the right part of the line is the words that tend to show up more in server toxic comments.The more right a word is from the line, the more distinct the word is for severe toxic comments. The words that shows up in the left part of the line is the words that tend to show up more in toxic comments. The more left a word is from the line, the more distinct the word is for toxic comments. For instance, in this visualization, we can observe that there is a toxic and severe toxic comment have high overlapping words. The words tend to cluster around the line. However, there are also several distinct word for each category. For instance, n-s, h-l and p-p are more distinct for severe toxic comments since it tends to show up less in toxic comments. Similar for other word clouds, based on the visualizations, we can see that the result is consistent with what we conclude above about the overlapping categories. Also, we do find out some unique words for each category which will be helpful later when fitting models. We also observe a large amount of common words that tend to show up in each category, for instance, f-k, s-k and f-g. We should be concern with these common words and avoid using them as an identifier for our model, since they lack representatives. Those common words tend to appear in most of the sub-categories, thus utilizing them in the model will make the model ignore the unique characteristics of each subcategory and therefore obscure our result.

Method

Model 1.1 Logistic Regression for Obscene Comments

We are first going to fit separate Logistic Regression for obscene and insult comments. Logistic Regression is the appropriate model to utilize when the dependent variable is binary. It is used to analyze the probability of certain events or class is existing. In our research, toxic comments contain 15294 observations while there are only 16225 nasty comments. Because of the overwhelmingly large number of comments, the result we get might not be typical and will probably lose the ability to characterize only toxic comments. Instead it resembles the characteristics of the whole nasty comments, which obeys our initial goal. Severe toxic comments, according to the word cloud visualization and high frequency word visualization, are highly overlapped with toxic comments. Threat comments only contains 478 comments and identity hate comments only contains 1405 comments. Since the number of comments for these sub-categories are relatively low comparing to the nasty comments. The sample size is too small for us to derive statistics that could represent the parameters in these two sub-categories. Therefore, we decide to only fit separate logistic regression model for obscene and insult comments.

From the visualization in last section, we observed top frequency words for obscene comments and several unique words of obscene comments. We then create several binary variables including top frequency words and unique words of obscene to record the appearance of those words in each nasty comment. The value of each variables are 1 if certain word are captured in the comments, otherwise the value will be 0. We finally decided to use n-e, f-k, s-k, b-h and c-t as our indicators. Below is the regression model weused.

Let Y_i be a comment in nasty comment data set, where i = 1,..., 16225.

Let π_i be the probability that the comment Y_i is classified as obscene.

Let N_i be an indicator that comment Y_i contains word "n-e".

Let F_i be an indicator that comment Y_i contains word "f-k".

Let S_i be an indicator that comment Y_i contains word "s-k".

Let B_i be an indicator that comment Y_i contains word "b-h".

Let C_i be an indicator that comment Y_i contains word "c-t".

$$Y_i \sim Bernoulli(\pi_i)$$

$$logit(\pi_i) = log(\frac{\pi_i}{1-\pi_i}) = \beta_0 + \beta_1(N_i) + \beta_2(F_i) + \beta_3(S_i) + \beta_4(B_i) + \beta_5(C_i)$$

The fitted model is:

$$log(\frac{\hat{\pi}_i}{1-\hat{\pi}_i}) = -0.0054 + 0.70(N_i) + 2.27(F_i) + 1.87(S_i) + 3.09(B_i) + 2.35(C_i)$$

We then create a confusion matrix to see how the model behaves, and also calculated the Classification Error Rate(CER) to check the overall performance of the model.

LogisticModel	ActualNotObscene	ActualObscene
PredictedNotObscene	4804	3682
PredictedObscene	3641	4094

The CER of this model is 45.14%. It means that our model could explain 54.86% of the observation correctly. It suggests that this model is doing an overall relatively good job in differentiating obscene comments and non-obscene comments, but still has potential to be improved. We then turned to fit a logistic regression for insult comments.

Model 1.2 Logistic Regression for Insult Comments

We follow similar steps in fitting logistic regression for insult comments as for obscene comments. We first create several binary variables including top frequency words and unique words of insult to record the appearance of those words in each nasty comment. The value of each variables is 1 if certain word are captured in the comments, otherwise the value will be 0. We finally decided to use s-d, f-k, s-k, b-h as our indicators. Below is the regression model we used:

Let Y_i be a comment in nasty comment data set, where i = 1,..., 16225.
Let π_i be the probability that the comment Y_i is classified as insult.
Let St_i be an indicator that comment Y_i contains word "s-d".
Let F_i be an indicator that comment Y_i contains word "f-k".
Let S_i be an indicator that comment Y_i contains word "s-k".
Let B_i be an indicator that comment Y_i contains word "b-h".

$$Y_i \sim Bernoulli(\pi_i)$$

$$logit(\pi_i) = log(\frac{\pi_i}{1-\pi_i}) = \beta_0 + \beta_1(St_i) + \beta_2(F_i) + \beta_3(S_i) + \beta_4(B_i)$$

The fitted model is:

$$log(\frac{\hat{\pi}_i}{1-\hat{\pi}_i}) = -0.14 + 0.57(St_i) + 1.81(F_i) + 1.11(S_i) + 2.09(B_i)$$

We then create a confusion matrix to see how the model behaves, and also calculated the Classification Error Rate(CER) to check the overall performance of the model.

LogisticModel	ActualInsult	ActualNotInsult
PredictedInsult	4445	3984
PredictedNotInsult	3903	3893

The CER of this model is 48.61%. It means that our model could explain 52.39% of the observation correctly. It suggests that this model is doing an overall relatively good job in differentiating insult comments and non-insult comments but can still be improved.

Model 2.1 Classification Tree for Insult and Identity hate comments

We then decided to use classification tree as our next method to further conduct our research topic. When facing data mining task that contains different classifications and then there exist clustering of data points, statisticians use decision tree models to resolve the problem (2015). There are two kinds of tree models: regression tree and classification tree. Regression tree model works for numerical variable while classification tree works for categorical variable. Classification tree make prediction by using clustering based on similarity of a group of observations. In the classification tree model, the data will split into partitions which is also known as the process of binary recursive partitioning(2009). After the split is conducted, the observations will be divided into different nodes by the splitting variable which is been predetermined. The splitting in each node will continue iterate until some stop conditions are reached(2012). There are several common stop conditions. The most common stop condition is that all leaf nodes are pure. There are also other stop conditions, for instance a given minimum number of observations in one node wasreached.

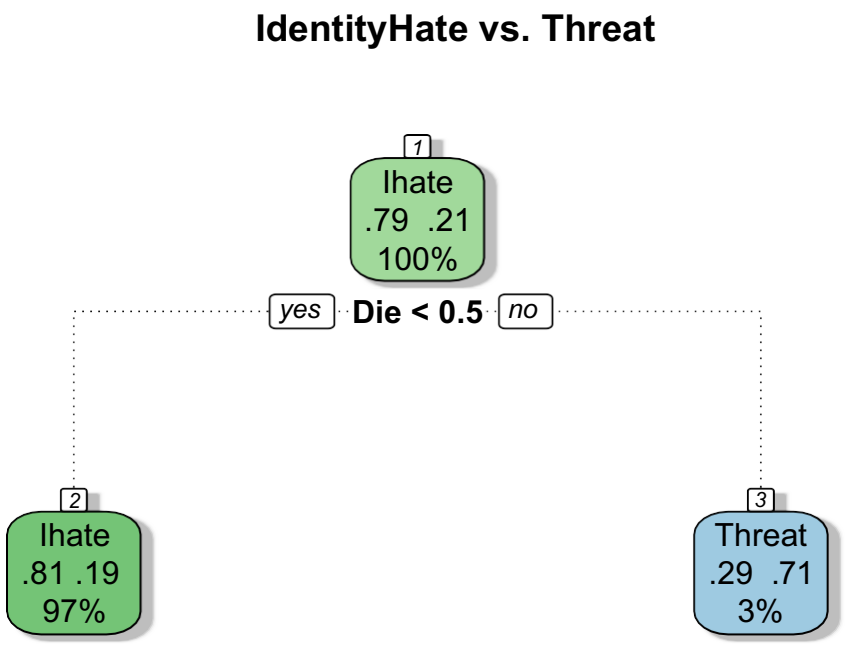
As we checked the overlap of each two categories before in previous section, the result suggested that toxic, obscene and insult tend to have high correlation with each other. These three categories have a high percentage of overlap. Therefore, we decided to combine these three categories together and create a new variable called TOI.We are not going to look at severe toxic comments. Accoeding to the top word frequency and word clouds, most of severe toxic comments are also toxic comments.However,when we closely look at TOI, we found that toxic comments has 15294 observations which accounts for 94.26% of the total nasty comments. We then decide to further split the TOI category by discarding toxic comments. Therefore, we create a new variable called OI, which only contains obscene and insult comments. While insult comments have 7877 observation and obscene comments has 8449 observations, the number of comments in two sub-categories are balanced.

We originally want to use classification tree to distinguish OI comments, identity hate comments and threat comments. Therefore, we also create categorical variable identity hate and threat and add them together with OI into the nasty data to identify OI comments, threat comments and identity hate comments separately.

However, classification tree requires a balanced number of observations in each category, and the number of OI comments are too much comparing to identity hate and threat comments, we then decide to only focus on identity hate and threat first. Therefore, according to the previous analysis on the word frequency and uniqueness, we pick word including d-e, f-t, f-k and n-r in identity hate and threats as potential identifiers.

We randomly choose 90% of the observations in Nasty Data as out training data and use the remaining 10% of the observations in nasty comments as our test data set. We extract all comments that classifiedas

identity hate and threat from the training data and fit a classification tree. The result is attached below.



According to the result, the identifier d-e is used to classify comments into identity hate and threat comments. Identity hate comments are defined by not including the specific word "di-" and the threat comments are defined by including the specific word d-e. In the identity hate cluster, there are 81% of identity hate comments and 19% of threat comments. In the threat cluster, there are 29% of identity hate comments and 71% threat comments.

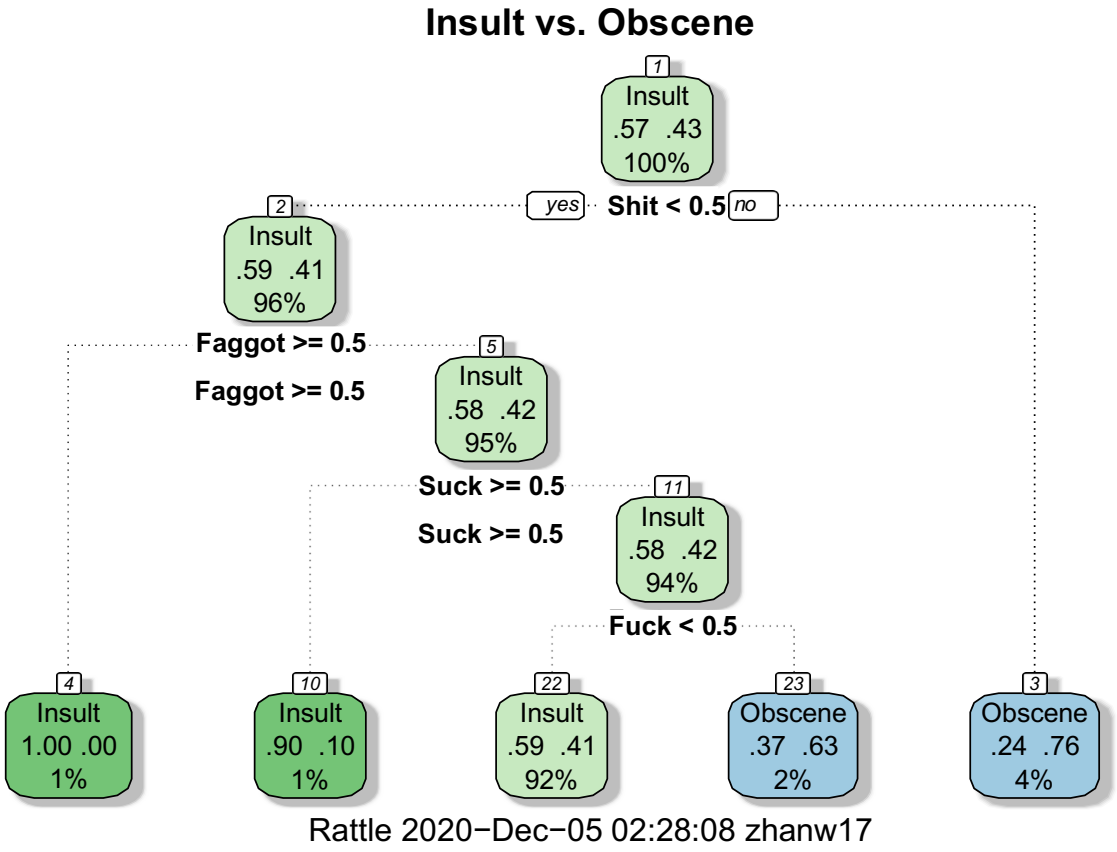
TrainData	ActualIhate	ActualThreat
PredictedIhate	1268	299
PredictedThreat	15	37

TestData	ActualIhate	ActualThreat
PredictedIhate	122	4
PredictedThreat	0	0

We then compute the CER for our model. The training CER is 19.4% and the test CER is 26.5%. The model's performance is overall acceptable given there is only one identifier being used. However, there are still some mis-classified comments suggesting that there is room for improvement. Also, there are no predicted threat or predicted identity hate comments for test data. It suggests that we are lack of observations for threat comments.

Model 2.2 Classification Tree for Obscene and Insult Comments

We then focus on insult and obscene comments while discarding toxic comments as we discussed previously. We then repeat the previous method: select words we are going to use as identifiers according to the visualizations of word frequency and word clouds, create identifies and fit a new classification tree. The model is shown below:



In this classification tree, we can see that f-k and s-t are two effective identifiers. According to our tree, the identifier s-t is used to classify comments into insult and obscene. The insult part is defined by not including the specific word "s-t". The threat part is defined by including the specific word "s-t". F-k serves as an identifier that further separates comments in cluster 2, which was already being separated by identifier Shit. The identifier insult is used to further classify comments into insult and obscene comments. The insult part is defined by not including the specific word "f-k". The obscene part is defined by including the specific word "f-k". Cluster 22 contains 59% Insult comments and 41% Obscene comments. Cluster 3 contains 24% Insult comments and 76% Obscene comments. According to the model, we can observe large amount of mis-calculation. The reason might attribute to the lack of observations. We then compute the CER for our model.

TrainData	ActualInsult	ActualObscene
PredictedInsult	469	321
PredictedObscene	15	38

TestData	ActualInsult	ActualObscene
PredictedInsult	49	39
PredictedObscene	0	0

From the confusion matrix of both training and testing data, we can calculate that the training CER is 39.86% and the test CER is 44.32%. However, in the testing data, we observe no predictions of obscene data. It was a big problem for the model and we believe the reason of this might attribute to lack of observations of obscene comments.

Conclusion and Further Work

For Model 1.1, our identifier N_i, F_i, S_i, B_i and C_i has a positive relationship with identifying an obscene comment. The expected log odds of a comments to be obscene is 0.69 higher if the comments contain n-g. The expected log odds of a comments to be obscene is 2.27 higher if the comments contain f-k. The expected log odds of a comments to be obscene is 1.87 higher if the comments contain s-k. The expected log odds of a comments to be obscene is 3.09 higher if the comments contain b-h. The expected log odds of a comments to be obscene is 3.09 higher if the comments contain c-t. All of the identifiers are statistically significant since their p-value are all approximately 0. We made confusion matrix to see how well the model performs. The CER of this model is 45.14%. It means that our model could explain 54.86% of the observation correctly. The model is doing an overall good job in differentiating obscene comments and non-obscene comments, but still has potential to be improved. In order to improve the model, we could select more unique word to obscene comments in the future as identifiers.

For Model 1.2, our identifier St_i, F_i, S_i and B_i have a positive relationship with identifying an insult comments. The expected log odds of a comments to be insult is 0.56 higher if the comments contain s-d. The expected log odds of a comments to be insult is 1.81 higher if the comments contain f-k. The expected log odds of a comments to be insult is 1.11 higher if the comments contain s-k. The expected log odds of a comments to be insult is 2.08 higher if the comments contain b-h.

All of the identifiers are statistically significant since their p-value are all approximately 0. We made confusion matrix to see how well the model performs. The CER of this model is 48.61%. It means that our model could explain 52.39% of the observation correctly. The model is doing an overall good job in differentiating obscene comments and non-obscene comments, but still has potential to be improved. In order to improve the model, we could select more unique word to insult comments in the future asidentifiers.

For Model 2.1 we fit a classification tree for threat and identity hate comments to find the words that could identify insult and identity hate comments. Our training CER is 19.4% and our test CER is 26.5%, suggesting an overall well performance. However, there are still mis-classified comments due to the lack of observations for these two categories.

For Model 2.2, we fit a classification tree for obscene and insult hate comments to find the words that could identify insult and identity hate comments. Our training CER is 39.86% and the test CER is 44.32%, suggesting an overall acceptable performance. However, there are still mis-classified comments. Also, the test CER are relatively high which means we have room of improvement.

One reason to explain the poor accuracy is that the lack of observations for obscene and insult comments. Therefore, we decide to stick on identity hate, threat, obscene and insult comments. Later on, we will try to include more identifiers. Currently we only have 5 identifiers for Model 1.1, 4 identifiers for Model 1.2, 4 identifiers for Model 2.1, and 7 identifiers for model 2.2. Creating and utilizing more identifiers may increase our accuracy. Moreover, we will consider to creating identifiers containing multiple words to increase the accuracy of out models. We will also try different models and method, for instance, random forest to see if better accuracy could be achieved.

Reference

Song, Y., & Lu, Y. (2015). Decision tree methods: Applications for classification and . . . Retrieved December 04, 2020, from https://www.researchgate.net/publication/279457799_Decision_tree_methods_applications_for_classification_and_prediction

Strobl, C., Malley, J., & Tutz, G. (2009). An introduction to recursive partitioning: Rationale, application, and characteristics of classification and regression trees, bagging, and random forests. Psychological Methods, 14(4), 323-348. doi:10.1037/a0016973

Hothorn, T., Hornik, K., & Zeileis, A. (2012, January 1). Unbiased Recursive Partitioning: A Conditional Inference Framework. Retrieved December 04, 2020, from https://www.tandfonline.com/doi/abs/10.1198/106186006X133933