Branched Classification of Hate Speech

Abstract

The detection of hate speech on online media platforms is a very important issue in the modern day. In this paper, we present a variety of methods for classifying comments as being hateful or non-hateful, using baseline models like Recurrent Neural Networks as well as transformer based architectures. The top transformer based models were then used to construct fine-tuned multi-class classifiers to predict the exact target of each hateful comment as well as the exact target of each racial hate comment. A final branching classification model was created using three BERT based transformers stacked together to first predict hate/non-hate and then the exact target for hateful comments. Unsupervised techniques were also used to discern relationships between hateful and non-hateful comments directly from the data.

1 Introduction

With the rise of social media over the last few decades, there has been a drastic increase of hate speech and hateful comments online. Developing methods of counteracting online hate is thus becoming an ever more important issue. Hateful comments can be classified in a number of different ways based upon whom the comment is targeted at, as well as the language used. For example, a comment can be offensive without being hateful. For the purpose of this paper we take hate speech to be language which is used to express hatred towards a targeted group of people. This can be categorised into derogatory or humiliating terms, insulting particular members of that group, dehumanisation and threatening language. This is a definition which has been adapted from (Davidson et al., 2017) and includes some of the key identifiers of hateful comments which are discussed in (Vidgen et al., 2020).

Our first aim is to use natural language classification models to detect whether a variety of comments are hateful or non-hateful. We will bench mark performance on a dynamically generated hate speech dataset which has been specifically designed to confuse models on edge case scenarios, (Vidgen et al., 2020). We hypothesise that transformer-based models with multi headed attention layers will perform better at this task than more classical natural language models such as Recurrent Neural Networks (RNNs) and Long Short Term Memory models (LSTMs).

Being able to detect hateful comments is a very important task which remains unsolved, due to the lack of usable data, poor generalisability and concerns of fairness in existing models. In this paper we sought to improve on this and take this analysis further by developing models that not only predict whether a comment is hateful or non-hateful, but try to predict the specific target of each of the hateful comments. In order to do this we compared the performance of the best performing models in two multi-class classification scenarios. The first of which specified whether a comment was targeted towards a particular race, gender, sexual orientation, religion, immigration status or other groups of individuals, while the second focused on discerning the specific race which was being targeted in the racial hate comments. Understanding the specific target of hateful comments can lead to better response mechanisms and helping victims of online abuse. Once each model has been explored we try and create a single branched model which first predicts whether a comment is hateful or non-hateful and if the result is hateful, predicts the specific target of hate. This model could now be deployed on social media platforms as a way of stopping the spread of hate speech, through identifying targets automatically, offering them the required support and encouraging the commenters from challenging their biases.

What makes a comment hateful can be subjective, hence making the task difficult. Unsupervised learning techniques were deployed to see if the structure and formulation of a sentence alone leads to a meaningful notion of what makes a comment hateful. If the data had been easily classifiable then we would expect it separate into meaningful clusters. Therefore, we hypothesised that we would be able to create clusters but they may not have necessarily corresponded to the labels used in multi-class classification.

2 Related Work

Hate speech detection has recently been an interesting research topic in natural language processing, with various techniques having been implemented. In the paper (Salminen et al., 2020), many different models such as Applied Naive Bayes, Support Vector Machines, Logistic Regression, XGBoost and a vanilla RNN were applied to a corpus of YouTube, Reddit, Wiki and Twitter data. They deduced that the best model used BERT-based embeddings with XGBoost classifiers. In (Biere et al., 2018) the focus turned to testing Convolutional Neural Networks (CNN) where the input layer transforms the data to look like an image to capture spatial correlations. They discovered in this paper that CNNs were outperforming SVMs, RNNs and LSTMs on a English Twitter dataset.

In (Davidson et al., 2017) they re-frame the problem to classifying tweets into three different categories. Hate speech, offensive language and neither. This was done by focusing on specific words and phrases in the lexicon associated with each class. Although this can lead to inherent biases it does explore the distinction between whether a person is making an aggressive comment or whether they are quoting Tim's abrasive rap lyrics. This demonstrates the importance of context when creating models.

More recent research has shifted to testing transformer-based models as they are easily parallelisable compared to RNNs, making it easier to train on large datasets. In (Mozafari et al., 2019), BERT-based transformer models were developed on the Twitter dataset explored in (Davidson et al., 2017), and these were then improved upon in (Mutanga et al., 2020) which found that the compact DistilBERT was outperforming other attention based models.

In (Gaydhani et al., 2018) n-gram features weighted with Term Frequency - Inverse Document Frequency (TF-IDF) values along with machine learning classification algorithms were used to classify Twitter comments into hateful, offensive and clean. Of the algorithms used, they found that Logistic Regression performed best using this approach.

In this paper, numerous different natural language models have been applied to benchmark and improve upon the task of hate speech detection. Although there has been extensive work on predicting whether a comment is hateful or non-hateful, this paper takes things further by trying to create a classifier which predicts the exact target of the hateful comments as well as the exact target of racial hate comments. This was formalised in a single model which can be used for more refined hate speech detection. Although, Latent Dirichlet Allocation (LDA) and Clustering have been explored in (Saini et al., 2020) and (Li and Liu, 2014) as a method for detecting hate, in this paper we use these unsupervised learning methods to see if hateful comments themselves are able to be divided further and the topic of the hate identified.

3 Dataset

The dataset used in this paper consists of synthetically generated hateful and non-hateful comments which were created through many rounds of adversarial training using human annotators. The dataset was specifically designed to address the common issue faced in the hate detection field which is the lack of available data used to train models. More details of the nature of the generation and why it was generated can be found in the following paper (Vidgen et al., 2020). The dataset consists of 40,632 different comments with 54% being hateful and 46% being non-hateful. Unlike most hate speech datasets there is a large proportion of hateful comments making it easier to train effective models. Although, this may lead to bias towards more false positives due to the relative infrequency of hateful comments in the general population on social media. It is also worth noting that the non-hateful comments were designed to try and deceive the models and are not representative of regular comments that may be found online.

The dataset also contains 40 different labels dictating the type of hate associated with each sentence. This was then distilled into six categories for the multi-class classifiers: Racial, Gender, Sexual Orientation, Religion, Immigration Status and Other. The relative frequencies of each class can be found figure 1.

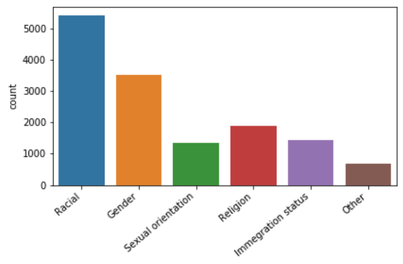

Figure 1: Histogram showing distribution of types of hate

When creating the racial hate classifiers the racial hate sentences were subdivided into the following 10 target types: Black, East Asian (including Chinese), South-east Asian (including Pakistan), Arabic, African, Hispanic, Eastern European (including Russian and Polish), Mixed Race, Indigenous, Traveller and an 'other' category (including comments defined as non-White and ethnic minority). Figure 2 shows the distribution of frequencies for these classes.

In order to run our models the data was split randomly into a training, validation and test set with proportions 80%, 10% and 10% respectively. For the multi-class classification tasks this was done on the reduced subsets of the data.

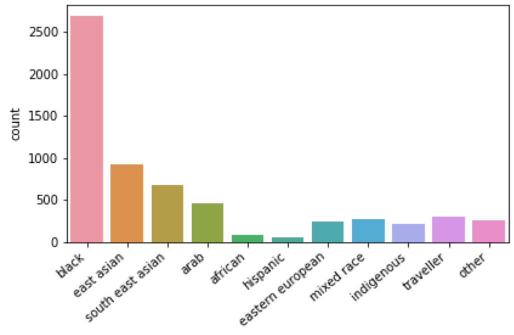

Figure 2: Histogram showing distribution of racial hate types

4 Methodology

4.1 Binary classification

In the first task, binary classification was performed on the hate speech dataset. Initial baseline models like Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long-short Term Memory (LSTM) and Gated Recurrent Unit (GRU) were implemented. We then sought to improve on these by using transformer-based models starting with the Bidirectional Encoder Representation Transformer (BERT) and then testing the more compact DistilBERT and more complex RoBERTa.

Before using the data we applied binary labels to the two classes, replacing the label "hate" with 1 and "non-hate" with 0. We then proceeded to tokenise the sentences and remove all punctuation and, for the baseline models, characters that are not part of the English alphabet. All sentence were converted to lowercase.

The CNN, RNN, LSTM and GRU all involved creating word embeddings using Word2vec on vocabulary based on news articles to produce 300-dimensional representations. Left padding was then applied to ensure all sentences were of the same length to be fed into the models. Left padding was chosen as we wanted the most important tokens to be fed into the models last to help preserve the meaning of the sentence. For all models the Adam optimiser was used. In the case of the CNN we then used the Binary Cross Entropy with Logits loss function and for the RNN, LSTM and GRU we chose the Cross Entropy loss function.

Word2vec embeddings are useful in many applications however there are still some shortcomings such as its inability to handle unknown words and the fact that there are no shared representations. For hate speech detection, it is important that word ordering is preserved as different word orderings strongly affect the target of each sentence. LSTMs and GRU models were tried due to their ability to capture more long term behaviours through the retention of information throughout each layer. However, in order to enforce this more strongly, we move on to using BERT models (Devlin et al., 2018). BERT is a deep transformer encoder, capable of processing long texts efficiently by using self-attention. As the name suggests, it is bidirectional and hence uses the whole sentence to understand the semantic meaning of each word. BERT embedding vectors generate 768-dimensional representations creating key, query and value vectors through linear projections. These vectors pass through a self-attention head which makes BERT "context-aware". This context awareness can be exploited to understand whether a particular comment is hateful or non-hateful. For this reason we believe the BERT-based models will easily outperform the baseline models with Word2vec embeddings. Other variants of BERT model such as RoBERTa (Liu et al., 2019) and computationally efficient variant DistilBERT (Sanh et al., 2019) were also implemented. All of the BERT-based models were implemented using HuggingFace APIs.

4.2 Unsupervised approach

Latent Dirichlet Allocation (LDA) LDA was used for the task of topic modelling, through building Dirichlet distributions such as a topic per document model and words per topic model. The aim was to investigate whether the model would be able to cluster the data into groups, by reviewing which words the model found contributed the most towards each topic. Pre-processing of the data involved the removal of all stopwords, lemmatization and tokenization before using both Bag-of-Words (BoW) and Term Frequency - Inverse Document Frequency (TF-IDF) methods to perform LDA. In the Bag-of-Words model, a dictionary is created, containing how many times a word appears in the training set. Tokens that appear in less than 15 but more than 0.5 documents are filtered out, keeping only the 100,000 most frequently used tokens. TF-IDF measures the relevance of a word to its document in a collection of documents, through multiplying two metrics - how many times the word appears in the document by the inverse document frequency of the word in a set of documents.

Clustering and Student-t Stochastic Neighbor Embeddings (t-SNE)

Another unsupervised approach is clustering similar comments together to generate a more concise and organized representation of the raw comment. An initial pre-processing (lemmatization) of the data was done and all comments were converted to lower case to avoid algorithms misinterpreting the same words as being different. In this approach, Word2vec embeddings were used to generate a 100-dimensional representation of the data. t-SNE is used to visualize high dimensional data by minimizing the Kullback-Leibler divergence between the joint probabilities of a low-dimensional embedding and the original high-dimensional data. Since the dimensionality is large and computationally expensive when computing t-SNE pairwise distances, an initial dimensionality reduction using principal component analysis (PCA) was performed. Principal components were selected to represent 95% of the variance of the features. Further, the cosine similarity metric was used to measure how similar two comments are irrespective of their size. In order to do this k-means clustering was applied with K set to being equal to 2,3,4,5 and 6.

4.3 Multi-class classification

In order to perform multi-class classification the three best models, BERT, RoBERTa and DistilBERT were used to see which of these performed best at detecting particular types of hate. To prepare for multi-class classification the data was categorised into six different categories based on the intended target of the hateful comments. To this end, we set "Racial Hate" to 0, "Gender-based Hate" to 1, "Sexual Orientation Hate" to 2, "Religious Hate" to 3, "Immigration Status Hate" to 4 and "Other" to 5. The same pre-processing steps as aforementioned were then applied in order to get the correct BERT representations. The models themselves then needed slight alterations in order to work for multiple classes. The loss functions were changed from Binary Cross Entropy with Logits Loss function to the Cross Entropy Loss function.

In order to create the multi-class classifier that focused on predicting the different types of racial hate, the same learning architectures were used except all data points in the racial category were selected, removing the rest. We then set "Black" to 0, "East Asian" to 1, "South-East Asian" to 2, "Arabic" to 3, "African" to 4, "Hispanic" to 5, "Eastern European" to 6, "Mixed Race" to 7, "indigenous" to 8, "Traveller" to 9 and "Other" to 10. These models were run for 10 epochs and the accuracy & loss at each epoch on the validation set were calculated. Confusion matrices were created from the test set predictions to develop a deeper understanding of misclassifications.

4.4 Branch-BERT Classifier

Finally, the previously mentioned binary and multi-class classification BERT models were combined in a decision tree-like structure. It first predicted whether a sentence was hateful or not and then went on to predict the type of hate for each of the hateful sentences. If the type of hate was deemed to be racial then the model would predict the target of racism.

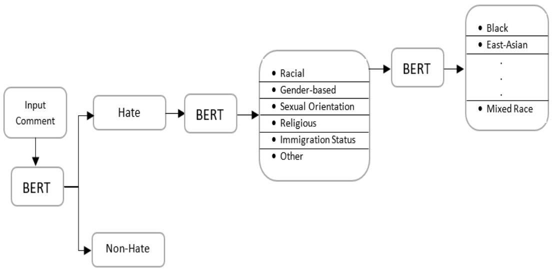

Figure 3: Branch-BERT Classifier

5 Results and Discussion

5.1 Binary classification

Model	Val acc	Test acc	F1 score
CNN	0.736	0.724	0.729
RNN	0.680	0.681	0.678
LSTM	0.756	0.748	0.747
GRU	0.755	0.754	0.753
BERT	0.874	0.886	0.883
RoBERTa	**0.893**	**0.893**	**0.892**
DistilBERT	0.840	0.851	0.857

Table 1: Binary classification metrics

For the binary classification task the performance of the CNN, RNN, LSTM and GRU were compared as well as the three BERT based models. Table 1 shows validation and test set accuracies alongside the F1 scores associated with each model. The F1 score is a metric which balances sensitivity, the per class best predictions, and precision, a measure of how relevant correct predictions are. The BERT-based models outperformed the baseline models in this task due to the architecture of the BERT models which allow for the semantic meaning of each sentence to be preserved. As expected the vanilla RNN model achieved the lowest performance as it is unable to capture long term behaviours within the sentences, and understand lexical links. Unsurprisingly DistilBERT was the lowest performing BERT model, potentially due to the fact it is more compact and so has fewer parameters. As expected, the RoBERTa model achieved the best performance here due to its increased complexity whereby it removes the Next Sentence Prediction task used in BERT's pre-training and instead masked tokens are changed during training epochs through dynamic masking. However, the RoBERTa model was computationally inefficient to train given the resources available and so we focused on the BERT model which provided the best balance between accuracy and computational efficiency.

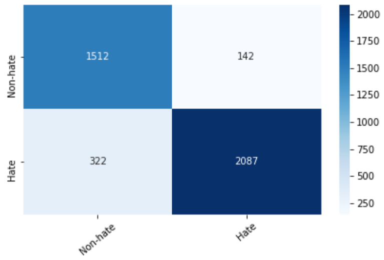

Figure 4: Confusion matrix for BERT Binary Classification

Example	Actual	Predict
It's as crazy as all other religions	Hate	Hate
this government is rubbish	Non-hate	Non-hate

Table 2: Correct Prediction Examples

Example	Actual	Predict
Wee Jimmy Krankie has a lot to answer for, Hideous f**king wh**e.	Non-hate	Hate
"each country belongs to the foreigner": jarring words from the pope	Hate	Non-hate
I'm pretty sure Hitler detested Indians.	Not-hate	Hate
disability is a result of illness	Non-hate	Hate
Ancient Chinese Mummies and Samurai have been found. Warning: not for the faint hearted.	Not-hate	Hate

Table 3: Incorrect Prediction Examples

Figure 4 shows the confusion matrix for the binary hate classification task for the BERT model. It suggests that on the whole the model is performing reasonably well, although it is noticeable that it has wrongly predicted hate when it is non-hate more often then the other way round, suggesting the model has a slight bias towards predicting hate, which is reasonable given that the dataset contains more hateful than non-hateful comments. Table 2 shows a few example of cases where the model is making correct predictions. Table 3 shows examples of situations where the model is making misclassifications. From the first example it is immediately clear that some of the cases in question are ambiguous and the model is clearly picking up on the right cues. The second example is a similar case whereby the exact meaning of the original sentence is unclear. Example three is a case whereby the model seems to be picking up on the sentiment but is unable to understand the difference between a statement of fact about someone else and an actual hateful comment being written. Finding models which are able to pick up on such nuances is a difficult task. The penultimate example represents situations in the dataset where the model appears to just be picking up on key words associated with a particular targeted group such as the word “disability” and so suggests the models are unable to truly understand the sentences in there entirety. The final example however is a classic case of where the model is going wrong and the prediction appears illogical. Figures 9 and 10 in the Appendix show convergence of accuracy and loss for the BERT model.

5.2 Unsupervised learning

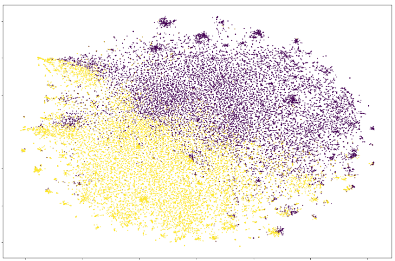

Figure 5: t-SNE Clustering of comments with K=2

K	2	3	4	5	6
Silhouette	0.157	0.334	0.195	0.102	0.056

Table 4: Silhouette scores for different values of K

Figure 5 is a t-SNE plot generated by performing K-means clustering, setting $K = 2$. Initially two clusters were used due to the original task of trying to distinguish between hate and non-hate. However, the two clusters provided little evidence of separating the data in a meaningful way as it was observed that both hateful and non-hateful comments were found sporadically within both clusters. Table 5 shows examples of sentences of the same class but assigned to different clusters. A similar trend was observed for different values of K, varied between 2 to 6. To find which K suited the best, a silhouette score was computed for each K. A silhouette score measures mean intra-cluster distance and the mean nearest-cluster distance for all the samples, where the highest value is 1 and lowest is -1. A silhouette score that’s close to zero indicates overlapping clusters which, from table 4, were observed in this scenario. The highest silhouette score **0.334** was obtained for the setting K = 3. The t-SNE plot for K = 3 (Fig 6) & K = 6 (Fig 15) can be found in the Appendix. Overall, the results were still inconclusive no matter the K value. For example, more than 5 anti-immigrant comments were found in all 3 of the clusters. The words used in most of these sentences are very similar and so the embeddings created through Word2vec were often extremely similar. It is worth considering, however, that the cluster assignments may be picking up on something different to the classes of hate defined in our multi-class classifier. Although, even looking from this perspective lead to inconclusive results. K-means clustering was not able to uniquely identify the different classes of hate and map them to distinguishable clusters.

Cluster 1	**Classification**
i hate when woman talk while they ’re on their period	Gender Hate
this area has a lot of mentally handicapped people . i think there is a home nearby .	Non-hate
Cluster 2	**Classification**
Women can never grow up, they are just useless in reality	Gender Hate
why did f**king covid come to this country and destroy everything	Non-hate

Table 5: Mixed Hate Type Examples

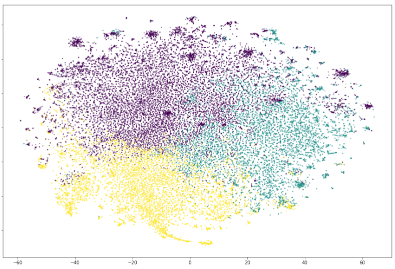

Figure 6: t-SNE Clustering of comments with K = 3

Continuing on from this, LDA was carried out on the dataset, using both Bag-of-Words and TF-IDF methods, outlined in the previous section. Different “topics” were discovered during training and comments such as those directing hate to people from different races and other countries, were tested. Topic Coherence is a metric that measures the degree of semantic similarity between highly weighted words in the topic. These measurements help differentiate between topics that are semantically interpretable and those which are results of statistical inference. The Topic Coherence for the Bag-of-Words and TD-IDF models were **0.285** and **0.263** respectively. This indicates that the BoW model topics are slightly more semantically interpretable than the TF-IDF model.

Some test comments were put through the LDA models, for example the comment “These women originally came from countries whose governments are a complete catastrophy”, obtains the following probability of topic scores along with the weighted contribution of each word in the topic found by the LDA model with BoW:

Score: 0.4615

0.02 “countri”; 0.019 “peopl”; 0.018 “year”; 0.014 “immigr”; 0.014 “want”

Score: 0.2576

0.125 “women”; 0.034 “f**k”; 0.018 “s**t”; 0.016 “like”; 0.016 “know”

As shown above, the topic most associated with this comment is xenophobia and this is related through words like “country”, “people” and “immigrant”. Therefore, this model has accurately predicted the topic of this comment. However, the topics themselves are not distinct and this could be a direct result of the fact that the comments themselves were difficult to cluster into types of hate. Although non-perfect there is definitely promising evidence that the LDA topics discovered are related to the types of hate which are being detected. Although each topic focuses on a lot of the curse words which are unrelated to type, there seems to be reasonable distinctions between different types of sentences.

Since the Topic Coherence and silhouette scores were low and led to topics which were not semantically interpretable, it did not seem logical to branch our classification using these unsupervised topics, but rather to use our intuition from the dataset.

5.3 Multi-class classification for different types of hate

For the multi-class classification task the three BERT-based models were implemented and the accuracies and F1 scores are shown in Table 6. From this table the BERT model achieved the highest accuracy scores, while surprisingly DistilBERT had a marginally higher F1 score. The performance scores were higher for this task as opposed to the binary classification task due to the fact it is a more refined problem and the classes could be more easily distinguished via specific words or phrases used against a particular group of individuals.

Model	Val acc	Test acc	F1 score
BERT	**0.930**	**0.931**	0.917
RoBERTa	0.921	0.918	0.916
DistilBERT	0.920	0.925	**0.919**

Table 6: Hate Types Multi-Class Classification Metrics

We chose the BERT model as it obtained the highest val and test accuracies and a strong F1 score. Figure 7 shows the confusion matrix for this task. What is immediately clear is that although there were misclassifications, they were well spread out and infrequent in comparison to the correct predictions. These correct predictions are highlighted in table 7 where the model was able to recognise both changes in characters as well as subtle tells such as the use of quotation marks to suggest sarcasm. From the confusion matrix it is clear that misclassifications happen more frequently when predicting the gender and race classes. The racial category contains many different groups of individuals and so to investigate this further, a more fine-tuned classifier was implemented.

Table 8 shows specific examples of sentences in which the model was misclassifying the data. The results of exploring these misclassifications show that there are multiple explanations for the incorrect classifications. The first two examples in the table are both cases where race and gender are being targeted in the sentence and so it seems plausible that the model is making mistakes. This suggests that perhaps the original labels themselves are slightly biased. In these cases it appears that the model picks up on the word “women” and predicts gender automatically without considering the wider context, suggesting a slight bias within the model. The third and fourth examples represent cases in the data where the model predictions seem accurate, and the assigned labels appear to be incorrect. Example three seems to be Islamophobic and therefore a religious hate comment, and similarly example four could be interpreted as anti-Semitic as it denies the Holocaust. Example five is a case where the overall sentence is ambiguous and difficult even for a human to interpret. The penultimate example, however, showcases where the model is making wrong predictions. It seemed to pick up on the word “’Women’” and predicted it to be a gender hate comment even though the true target was Muslims. In the final example we have a case where the words used in the sentence are not from the English language and so although the BERT tokeniser is able to pick up on some of these instances it will occasionally assign them the “unknown” token. The BERT model seems to be doing a very reasonable job at making predictions and even the misclassifications are logical more often than not.

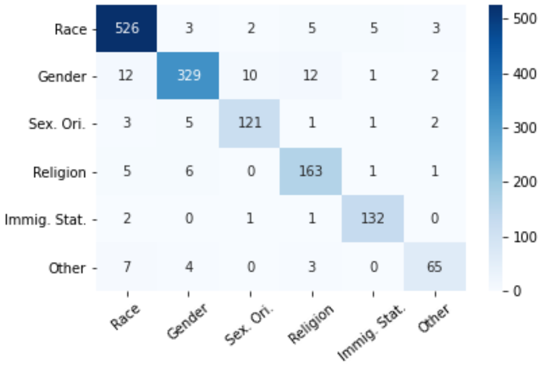

Figure 7: Confusion matrix for BERT hate types multi-class classification

Example	Actual	Predict
4r4bs R n33dy	Racial	Racial
No one loves fucking rude rope heads	Gender	Gender
i know so many trans ’men’	Gender	Gender

Table 7: Correct Prediction Examples of Hate Types

Example	Actual	Predict
asian women are filth? You are entirely right!	Racial	Gender
she looks ugly so neither makeup nor plastic surgery can help, all russian women do tho	Racial	Gender
muslims = egregious	Gender	Religion
well there was no Holocaust so the whole thing is a lie regardless of what you think	Other	Religion
its obvious, they destroy everywhere with beds like hotels, their nests, and then expect someone to clean it up.	Racial	Gender
hey oh God Muslim Goys Taking Our Women	Religion	Gender.
g!ps!3s keep out	Immig Stat	Sex Ori

Table 8: Incorrect Prediction Examples of Hate Types

5.4 Multi-class classification for different types of racial hate

In this section, we look at the results of fine-tuned racial hate classifiers which can be used in cases when the multi-class classifier predicts a comment that is of a racial nature. Table 9 shows the resulting validation and test set accuracies alongside the corresponding F1 scores. In this case, the DistilBERT model achieves the highest validation accuracy but lowest F1 score and test accuracy. The RoBERTa model achieves the highest test accuracy and F1 score, similarly to in the binary classification case, again highlighting the power of using more parameters. However, again the BERT model manged to provide the best balance between high metric scores and computational efficiency is why this model was used for our branching BERT classifier. All models performed very well on this task which is even more refined than in the previous task.

Model	Val acc	Test acc	F1 score
BERT	0.951	0.935	0.895
RoBERTa	0.948	**0.947**	**0.904**
DistilBERT	**0.953**	0.927	0.880

Table 9: Racial Hate Types Multi-Class Classification Metrics

Figure 8 shows the confusion matrix for the BERT racial multi-class classifier. The confusion matrix is very sparse with relatively few misclassifications indicating that the classifier is doing a reasonable job. Examples of correct predictions can be found in **??**. However, it is noticeable that after removing the unrelated data points, there is less data available for this task. For example, for the Hispanic comments there were only 6 examples in the test set to classify, suggesting that although the hit rate is high, there is insufficient evidence to back up true claims of model performance. Similar to the multi-class classification task, table 11 shows examples of situations where the model is making incorrect predictions. The first example is a case where there are many correct labels like South East Asian, Black, non-White that could be associated with the sentence and so it is unsurprising that the model made a few wrong predictions here. The second and third examples the predicted answer could easily be justified as being the correct answer. The fourth example here is an example where the overall sentence was ambiguous, it is very difficult for the model to understand that a specific person is related to a specific race and so it is anticipated the model failed in this case. The penultimate example is one of the few cases in the 40 misclassifications where the model is wrong with little justification. The final example is then another case where non-English characters lead to the model struggling to make correct decisions.

On the whole it seems the BERT model is doing very well in terms of making predictions. The majority of misclassified sentences where given a label that makes logical sense. For example the Black and African comments are often mistaken as well as Black and Other. These predictions however can sometimes be correct. However, there seems to be a bias towards predicting the Black category when the model is uncertain, which reflects the imbalance in training data as seen in Figure 2.

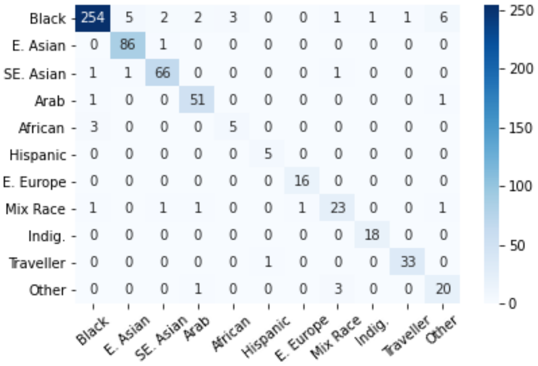

Figure 8: Confusion matrix for BERT racial hate multi-class classification

Example	Actual	Predict
He's annoyed because he realised his mutt baby will not look anything like him	Mix Race	Mix Race
the menace of burritos and biryanis is absolutely terrifying	SE. Asian	SE. Asian
P0l3s r d0gs	E. Europe	E. Europe

Table 10: Correct Prediction Examples of Racial Hate Types

Example	Actual	Predict
I guess as long as the fascists attack the brown, black and 'far left' people they'll never lose support from me and other whites.	Other	Black
I endorse imprisoning those who are originally from Africa	Black	African
england is not for you as your family is not white	Other	Black
I'm voting for senator blackface Comala Harris	Mix Race	Black
Those Japs would rather nuke themselves than address their hellish work culture.	E. Asian	Black
!nd!@ns @r3 basically mites and lice!	SE Asian	Black

Table 11: Incorrect Prediction Examples of Racial Hate Types

5.5 Branch-BERT Classifier

From the results we have explored it seemed plausible to create a final model which used all three classifiers to make fine-tuned predictions for new sentences, as each model performed strongly. BERT models were used because they are far more computationally efficient than RoBERTa but obtain comparable performance. Using the structure defined in 3, a Branch-BERT Classifier was formed which was able to take a sentence as an input and make predictions. Table 12 shows specific examples to highlight the model's performance. The predictions made from this model were successful with the ability to pick up on subtle notes within sentences. This model brings together the successes from the individual classification models.

Example	Predict
Chinese people are nice	Not-Hate
Chinese people are not nice	Hate Racial Hate E. Asian
They are stealing our jobs. From hard working americans.	Hate Imm. Stat.
i hate everything that your kind stand for	Hate Gender

Table 12: Branch-BERT Examples

6 Conclusion

The goals of this research were to detect and classify hate speech comments to help mitigate the toxic nature of online social media platforms. This began by using natural language models to try to understand whether a specific comment was hateful or non-hateful. The idea then developed into the formulation of more fine-grained classifiers used to detect different types of hate and further to detect different types of racial hate. This is an extremely important task as no group should feel victimised or unsafe on online platforms. Being able to understand the exact target is also an important question as this can help with developing mechanisms to challenge those producing such comments, as well as knowing who may need extra support off the back of experiencing such comments and encouraging the commenter to question their biases. However, none of these tasks are particularly easy due to the fine line between a comment being hateful towards a particular group of people and a comment being hateful in tone but not necessarily toward a particular individual or group of individuals. The dataset consists of many sentences such as "this parliament is rubbish" which can be considered to be negative language but is non-hateful. This means that the models produced need to learn more subtle lexical notes than merely specific key words.

The models produced in the binary classification task and both of the multi-class classification tasks are successful in making predictions. Although errors do occur, they are often due to ambiguous sentences where the prediction by the model makes logical sense. All the BERT-based models were able to perform well due the fact that they retain more information through attention layers. Both the regular BERT and RoBERTa models get the highest results, although due to the fact the BERT model requires far fewer parameters, this model was used in order to create a single hate detection classifier. In the process of trying to understand the data in more detail and seeing if we could generate cluster assignments independent of those given in the dataset, various unsupervised learning tasks were performed. Although LDA seemed to produce somewhat comprehensible topics, they were not semantically interpretable and the different clusters produced from t-SNE were intertwined. This highlights the difficulty of hate speech detection, especially when using pretrained Word2vec models which were unable to recognise words not written in English characters.

Although there is still work to be done to improve on this task, we provide some useful benchmarks in this paper as well as a unique model, Branch-BERT, which performs end to end hate speech detection. Very little literature tries to perform multi-class classification on hate speech data and so our novel approach gives more detailed classifications which can be used to make social media platforms safer.

7 Future Work

The methods explored in this paper only scratch the surface of the tasks that can be done in the hate speech detection domain. A big issue is the lack of usable datasets with high quality labeled data. To take this research further, the models presented should be applied to differing datasets such as Twitter, YouTube and Wikipedia, to compare performance against other papers and ensure the model is fit for online data. The dataset used here focuses on distinguishing between hateful and aggressive/offensive comments. Further preprocessing may also be required to ensure that sentences which include non-English characters, multiple instances of the same letter and text speech are well defined in training.

The Branch-BERT model presented here can also be improved upon by adding further branches which try to classify different types of hate. For example, the Gender based category includes hateful comments towards women as well as towards transgender people and the Immigration Status category can be further divided into subcategories such as refugees and immigrants.

A task we began to explore but needs further insights is the idea of using a Variational Autoencoder (VAE) to perform style transfer from hateful comments to non-hateful comments. This can be used to discourage people from writing a hateful comment by suggesting to them an alternative non-hateful remark. The method(John et al., 2018) works by using the VAE to learn both a content and style embedding and then create a new sentence based off of the content representation and the known style representation we are wishing to replicate. However, based on the unsupervised learning clustering tasks, specifically using the K-Means algorithm with $K = 2$, we came to the conclusion that this task would be challenging as the model was unable to find meaningful distinguishable clusters between the hateful and non-hateful comments. The problem is further ill-defined due to the fact that the hateful part of a comment both defines its style and its content and so altering the style without altering the content is very difficult. With deeper exploration into how to represent style as well as examples of the kind of sentences that we would like to generate would make this an interesting and engaging area of research. Style transfer has primarily been used for tasks like altering sentiment and so exploring its usefulness for hate speech would be a new problem.

References

Shanita Biere, Sandjai Bhulai, and Master Business Analytics. 2018. Hate speech detection using natural language processing techniques. *Master Business AnalyticsDepartment of Mathematics Faculty of Science*.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Aditya Gaydhani, Vikrant Doma, Shrikant Kendre, and Laxmi Bhagwat. 2018. Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. *arXiv preprint arXiv:1809.08651*.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2018. Disentangled representation learning for non-parallel text style transfer. *arXiv preprint arXiv:1808.04339*.

Gang Li and Fei Liu. 2014. Sentiment analysis based on clustering: a framework in improving accuracy and recognizing neutral opinions. *Applied intelligence*, 40(3):441–452.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. 2019. A bert-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications*, pages 928–940. Springer.

Raymond T Mutanga, Nalindren Naicker, and Oludayo O Olugbara. 2020. Hate speech detection in twitter using transformer methods. *International Journal of Advanced Computer Science and Applications*, 11(9).

Yash Saini, Vishal Bachchas, Yogesh Kumar, and Sanjay Kumar. 2020. Abusive text examination using latent dirichlet allocation, self organizing maps and k means clustering. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 1233–1238. IEEE.

Joni Salminen, Maximilian Hopf, Shammur A Chowdhury, Soon-gyo Jung, Hind Almerekhi, and Bernard J Jansen. 2020. Developing an online hate classifier for multiple social media platforms. *Human-centric Computing and Information Sciences*, 10(1):1–34.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Bertie Vidgen, Tristan Thrush, Zeerak Waseem, and Douwe Kiela. 2020. Learning from the worst: Dynamically generated datasets to improve online hate detection. *arXiv preprint arXiv:2012.15761*.

8 Appendices

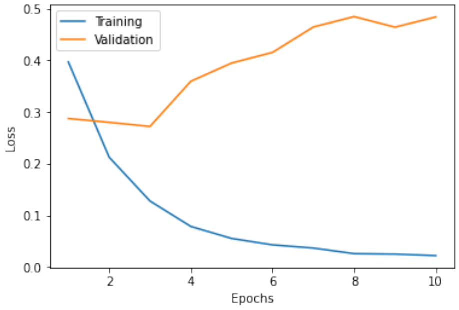

Figure 9: Loss plot for BERT binary model

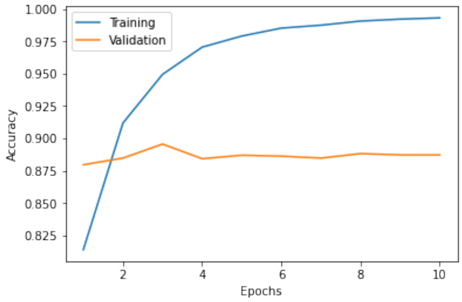

Figure 10: Accuracy plot for BERT binary model

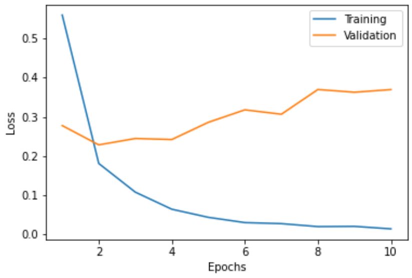

Figure 11: Loss plot for BERT multi-class model

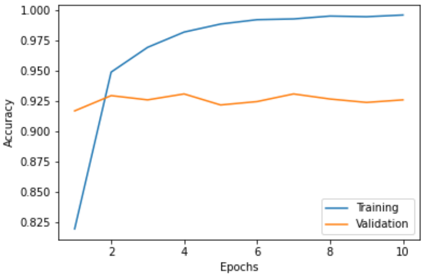

Figure 12: Accuracy plot for BERT multi-class model

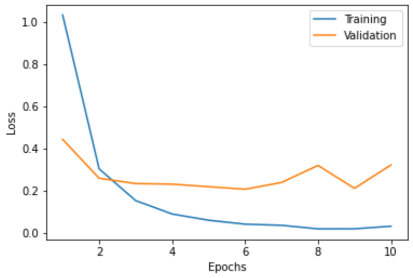

Figure 13: Loss plot for BERT race multi-class model

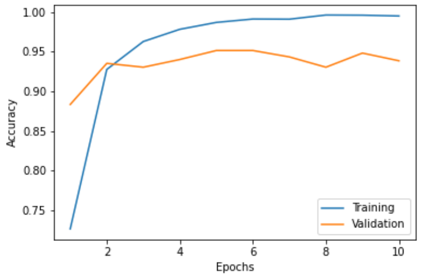

Figure 14: Accuracy plot for BERT race multi-class model

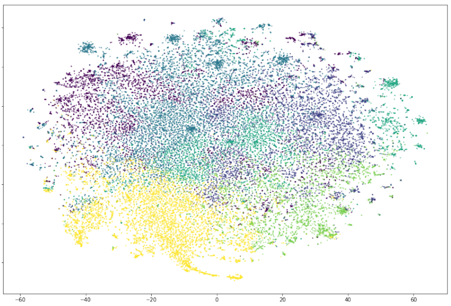

Figure 15: t-SNE Clustering of comments with K = 6