Toxic Speech Detection

Samayamanthula Lokesh Kumar, Tummalapalli Sree Rama Vijay and M Baskar

May 17, 2023

TOXIC SPEECH DETECTION

S. Lokesh Kumar[1],T.Sree Rama Vijay[2] and Dr.M.Baskar[3]

[1] SRM Institute of Science and Technology, Chengalpattu, Tamil Nadu,603203, India

[2] SRM Institute of Science and Technology, Chengalpattu, Tamil Nadu,603203, India

[3] SRM Institute of Science and Technology, Chengalpattu, Tamil Nadu,603203, India

Abstract: People on the internet nowadays frequently post comments or write statements that are quite nasty and deemed abusive language on social media platforms and in meetings, for example. Detection of the system is extremely challenging to solve manually, so we must design an automatic approach. Once restricted to verbal communication, hatred has rapidly spread via the Internet as more people have access to social media and online discussion groups, they are using them to disseminate hateful messages. Numerous nations have passed laws to curb the dissemination of hate speech online. In light of the platforms' repeated failures to curb the spread of hate speech, they hold the companies responsible. However, as internet information grows, so does the propagation of hate speech. Human monitoring of hate speech on online platforms is not only impractical but also prohibitively expensive and time-consuming because of the massive amounts of data that are involved. Therefore, it is essential to automatically monitor online user content for hate speech and remove it from online media. Since many contemporary methods are not easily interpreted, it can be puzzling to learn why the systems have reached their verdicts. This article suggests using the Support Vector Machine (SVM) and the Naive Bayes algorithms to automatically detect hate statements in online discussions. Despite being simpler and giving findings that are more easily interpretable than other methods, this method attained near-state-of-the-art performance. When this method was put through some empirical testing, SVM achieved an accuracy of almost 99% on the test set, while NB only managed 50%.

Keywords: Machine learning SVM, decision tree, f1score, classification

Introduction:

It is regrettable that hate crimes are not a new phenomenon in our culture. However, internet platforms like social media and messaging apps have started to play a bigger role in hate crimes. There have been a number of recent hate-motivated terror acts, and all of the suspects had long online histories of posting hateful material. In some circumstances, social media can play a more important role; for example, video footage from the suspects in the 2019 terrorist attack in Melbourne, New South Wales, Australia was aired live on Facebook.

The vast reach of the internet's many communication platforms like social media gives users a platform for unfettered, often anonymous expression. While the power to express oneself freely is a fundamental right that ought to be respected, promoting and promoting hate against some other group is an abuse of this freedom. In the United States, for instance, the American Bar Association maintains that, absent an express incitement to violence, hate speech is authorised and protected under the First Amendment. As a result, several websites, like Facebook, YouTube, and Twitter, consider hate speech to be detrimental and also have regulations in place to delete offensive speech content. There is a tremendous desire to research the automatic identification of hate speech due to the growing concern from society as well as the increasing prevalence of hate speech on the Internet. By automating its detection, it is possible to reduce the dissemination of nasty content.

The identification of hate speech, on the other hand, might be difficult. First, there are differences regarding the definition of hate speech. This means that, depending on their definitions, some content may be labelled incitement to violence by some however not by others. We begin by taking a look of differing perspectives, focusing on the numerous variables that contribute to offensive speech. We are not comprehensive, neither may we be, because new meanings emerge on a regular basis. Our goal is just to demonstrate differences and the ensuing concerns.

The effectiveness of offensive speech detection techniques is challenged by the use of differing ideas. Existing datasets define hate speech differently, leading in statistics which not only have been sourced from multiple sources and also include different information. Due to this, it may be difficult to identify the presence of offensive speech with precision. The next section will focus on the various datasets that can be utilised in the training and evaluation of offensive speech detection techniques. Nuances and complexities in language provide further difficulties for computer aided hate speech identification, which is itself dependent on definition.

Despite these distinctions, current approaches to identifying hate speech in textual content have demonstrated encouraging results. It is possible to utilise machine learning methods to spot instances of hate speech in the text with the given solutions. One potential drawback is that the decisions made by these methods may lack transparency and be hard for people to understand. This is a realistic concern because appeals processes for speech-censoring systems are typically human-intensive. To overcome this issue, we offer a new method for classifying hate speech that provides a better explanation of the classification decisions and demonstrates that it outperforms existing methods on particular datasets. Some current methods draw on information from the outside world, such as a hate speech lexicon. Although this strategy has the potential to yield positive results, it is problematic because it necessitates the constant upkeep of these sources. Our method does not rely on other sources and reaches a respectable level of accuracy. We address these issues in the subsequent section.

Related Works:

There is not an adequate or understandable explanation of what HS is. The comment that caused harm is called to as HS. Despite the fact that the HS is typically called to as an unpleasant statement, few studies used the terminology offencive speech in their research to recognize HS on social networking sites, researchers used two approaches: I a traditional machine learning methodology as well as (ii) a deep learning-based technique. The subdivisions that follow go over existing research that uses these two techniques.

A. Predicting Haterativity in Online Forum Discussions(ML)

Information for Warner and Hirschberg's investigation came from two online sources: Yahoo! and also the American Jewish Committee. Optimum value of 0.68, 0.60, 0.64, and 95% were achieved for precision, recalls, and F1-score, respectively, while employing the SVM light classifier. With the use of a Naive Bayes classifier, Kwok and Wang were able to categorise the tweets using the Bag-of-Words feature extraction technique.

Using a 10-fold cross-validation configuration, the model achieved 76% accuracy in the best instance. They stated that the BOW model is inadequate for classifying tweets relevant to HS. To achieve this accuracy, they just used the uni-gram feature, and they speculated that adding the bi-gram and emotion score of tweets to the feature set could improve performance even further.

Burnap and Williams gathered 450,000 tweets for their study. Word features from tweets were collected as N-grams (1-5) and then categorised using the supervised model. Classifiers such as voted ensemble classifier, Support Vector Machines (SVM), and Bayesian logistic regression were analysed. The Voted Ensemble Classifier performed best, with scores of 0.89, 0.69, and 0.77 for accuracy, recall, and Fl-scores, respectively. Waseem and Hovy offered up a 16000 labelled dataset for the purpose of HS detection. From tweets, uni-, bi-, tri-, and quad-gram characteristics were retrieved. The F1-scores for the Gender, Length, (Gender+ location), and (Gender+ location+ length) feature sets were 73.89%, 73.66%, 73.62%, and 73.47%, respectively, when using the logistic regression classifier with 10-fold cross validation.

B. Deep Learning Methods to Offensive Speech Identification

Djuric et al. created a model for recognising HS in user comments. They used continuous bag of words (CBOW) and paragraph2vec to capture comments in a constrained space. These features were then fed into a binary classification to label the comments as hostile or neutral; paragraph2vec reached the greatest possible area under the receiver operating characteristic curve (AUC) of 0.80. Tweet classification was accomplished by Park and Fung using logistic regression and a CNN algorithm. They carried out their research using both deep learning models and conventional machine learning classifiers, and they found that the two models worked better together than they did separately.

Zhang et al. introduced a model for HS identification that merged a convolutional neural networks (CNN) with a Gated Recurrent Unit network. Using seven publicly available datasets, they found that their model achieved an average Fl-score 14% higher than the other six models. This method use a combination of sequence and semantic analysis to provide meaningful results from minimal material. Kamble and Joshi's work on tweets with a mix of English and Hindi for HS detection was a big step forward. They were embedded using the huge dataset of mixed-language data that was used to build the model. According to the findings of the experiments, the newly developed code-mixed embedding performed significantly better than the which was before word embedding. Several classifier models were used in the experiment, included SVM, Random Forest, CNN-1D, LSTM, and Bi-LSTM models. The CNN-1D model achieved the best results, scoring 83.34 out of 100 for precision, 78.51 out of 100 for recall, and 80.85 out of 100 for Fl-score accordingly.

Researchers built a plethora of models based on machine learning and deep learning in order to solve safety issues regarding Twitter. Utilizing Bag-of-Words, n-gram, tf-idf, and one-hot encoding characteristics, they were able to create a more accurate model. To successfully forecast HS tweets, traditional machine learning-based algorithms required considerable showcase building, which is a time-consuming task and a separate research subject. The high rate of misclassification may be traced back to the perceptron model's reliance on one-hot encoded input, which does not maintain the tweets' meaning. Only a small number of academics utilised this model to make their predictions about the HS. To get over these limitations and improve prediction quality, this research uses a model based on convolutional neural networks.

Proposed Methods:

The process of logistic regression can be thought of as a supervised classification method Classification problems involve an output variable, y, that takes on only a small set of discrete values based on a large set of input attributes, X. Contrary to common belief, logistic regression can be thought of as a type of regression model. Data items' classification into the "1" category is predicted using a regression model built using the sigmoid function is used in logistic regression to model the data, just as the linear assumption is used in linear regression.

Support Vector Machines (SVM) is a well-known supervised learning method that may be applied to problems of classification and regression. However, classification problems in the domain of Machine Learning are its primary use.

The goal of the support vector machine (SVM) algorithm is to locate the best line (or decision boundary) for classifying n-dimensional space, making it simple to assign new data points to the correct class in the future. The most appropriate boundary is a hyperplane.

The hyperplane is formed by a collection of extreme points and vectors, some of which are chosen using a support vector machines (SVM). Support vectors, the most extreme cases, inspired the development of this method, which is why it is referred to as "Support Vector Machine”. View the accompanying illustration to understand how a decision boundary (or hyperplane) can be used to divide a set of items into two categories.

Hyperplane: For a set of data in n dimensions, there may be multiple feasible boundaries (decision boundaries) that can be used to partition the data into meaningful subsets. However, it is important to identify the one that best aids in data classification. The optimal boundary, often called the SVM hyperplane.

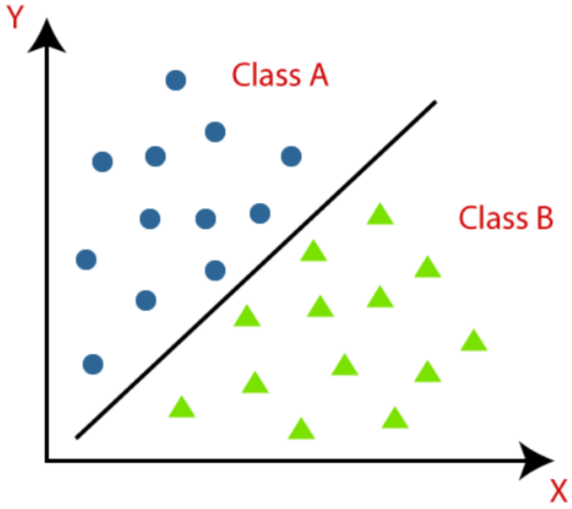


Fig 1:SVM Classification

If there are only two features (as depicted in the image), the dataset has a single-dimensionality, and the hyperplane is a straight line. Also, if there are three characteristics, the hyperplane will only have two dimensions. A hyperplane with the largest possible margin (the greatest possible separation between data points) is always constructed.

Support Vectors:

The term "Support Vector" is used to describe the set of data points or vectors that are most in close proximity to the hyperplane and have the most influence on where the hyperplane lies. For this reason, we refer to this class of vectors as "support vectors," as they are essential to maintaining the hyperplane.

As a supervised learning method grounded in Bayes' theorem, the Naive Bayes algorithm is put to use in situations where there is a need to make a determination between classes. It finds its most common application in text classification problems with a large and complex training dataset. The Naive Bayes Classifier is one of the classification algorithms that is both the easiest to use and one of the most successful. It is used in machine learning to construct prediction models that are quick and accurate. Since it is a classification technique, it determines the existence of an object by weighing the odds. The Naive Bayes Algorithm has a wide range of applications, some of which include the filtering of spam, the mining of opinions, and the classification of content.

You may learn how the Naive Bayes Classifier works by looking at the code below:

Let's pretend we have access to a dataset detailing the day's weather and a variable named "Play" that represents our intended outcome. Consequently, we need to use this data set to determine whether or not to play on a certain day based on the weather forecast. Therefore, the following procedures must be taken to address this issue:

1. Create a set of frequency tables utilising the given dataset.
2. Determine the probabilities of the specified attributes to generate a Likelihood table.
3. Proceed by determining the posterior probability using Bayes' theorem.

Architecture Diagram:

Data Gathering: The information was collected by us from public sources. Including comments and tweets from various online sources, categorised.

Data Modelling: The data were then modelled into labels, and the model was trained using the labels as input.

Classification: Machine Learning algorithms are performed on the complete data set before it is stored, and then the data is categorised into a variety of label's.

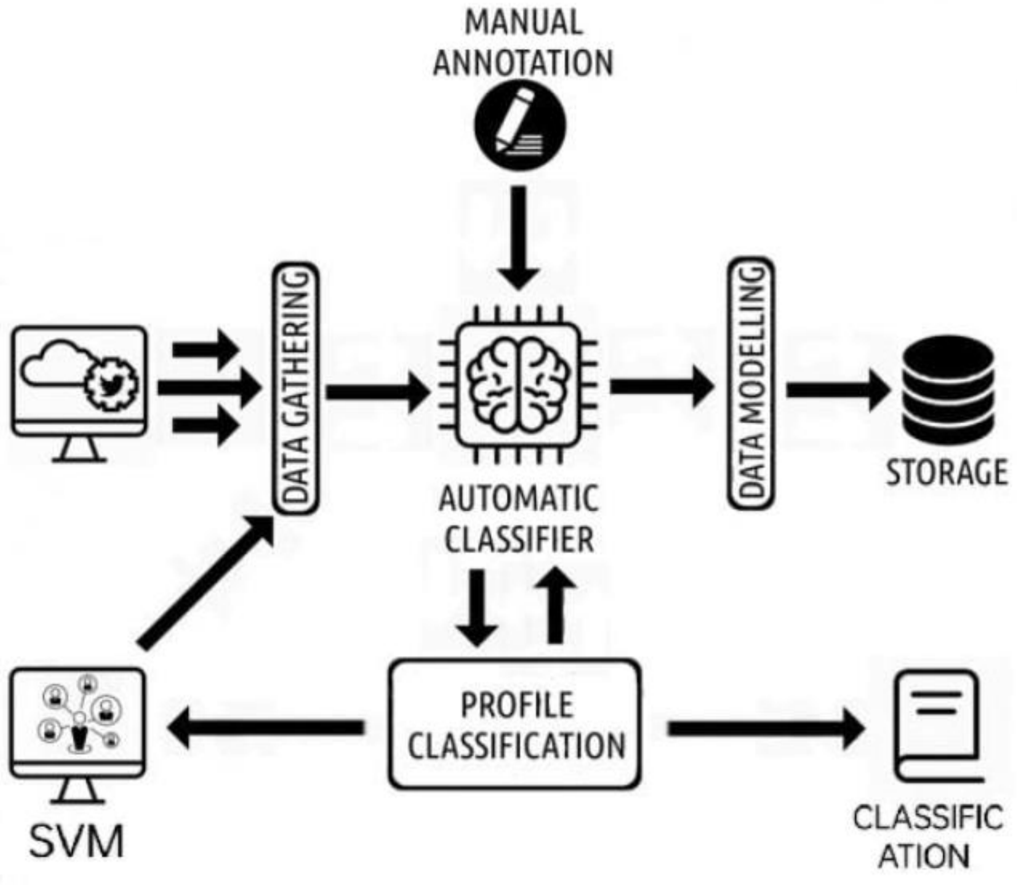


Fig 2: Architecture Diagram of Toxic Speech Detection

Results and Discussion:

With the exception of F7, for which the "hate" label's precision, recall, and a zero(0) f1 score indicates a perfect performance., the logistic regression approach functions beautifully across all feature sets. The Random Forest classifier performs pretty well for F1 and all other feature sets; however, when tf-idf scores are omitted from the category, its productivity is drastically diminished. While the overall performance of the Naive Bayes classification technique for labelling tweets as violent, provocative, with either is found to be less impressive, it performs significantly better with parameter F7 than with any other feature. It appears that the SVM classifier is consistent across all features except for F4 and F7. Based on the graphs presented above, our analysis reveals that the TF-IDF scores, denoted by F1, are the most significant feature, Since they help to a more precise categorization of hate speech, they are included. The sentiment scores appear to be an important factor in separating hate speech from offensive language. When eliminated from the feature set, Doc2vec columns do not contribute much to classification accuracy. After examining each of the graphs presented above, it is quite evident that Random Forest is the most successful.

Accuracy:

When evaluating classification models, accuracy is a metric used to quantify how often the model correctly predicts a class.For example, if you correctly predict 90% of the time, your accuracy is just 90%.When you have a classification with a balanced distribution of classes, accuracy becomes a valuable metric to employ.

$$\text{Accuracy} = \frac{\#\ of\ correct\ predictions}{\#\ of\ total\ predictions}$$

Precision:

The F1 Score's accuracy component comes first. Additionally, it can be utilised on its own as a metric for individual machine learning applications. The formula is as follows:

$$\text{Precision} = \frac{\#\ of\ True\ Positives}{\#\ of\ True\ Positives + \#\ of\ False\ Positves}$$

This formula can be understood in the following manner. When evaluating the likelihood of a positive outcome, accuracy measures how close to the mark predictions actually are.

A not precise model may find a lot of the positives, but its selection method is noisy: it also wrongly detects many positives that aren’t actually positives.A "pure" model has a high degree of accuracy; it may not find every positive, but the ones it does identify as such are quite likely to be right

Recall:

The F1 Score includes a component for memory, and recall itself is a useful metric for machine learning .In this example, we can see the formula for memory recall:

$$\text{Recall}= \frac{\#\ of\ True\ Positives}{\#of\ True\ Positives+\#\ of\ False\ Negatives}$$

This equation can be understood in the following way. How many good things did the model manage to find among all the good things there are?

An effective model will have a high recall, meaning that it will correctly identify the vast majority of positive examples in the data. However, it may also mistakenly classify some negative cases as positive.If the model has poor recall, it won't be able to identify most (or even most) of the true positives.

F1 score:

Accuracy and memory are averaged into a single metric known as the F1 score.The harmonic mean, as a quick refresher, is a substitute for the more standard arithmetic mean.It's a common tool for figuring out averages.

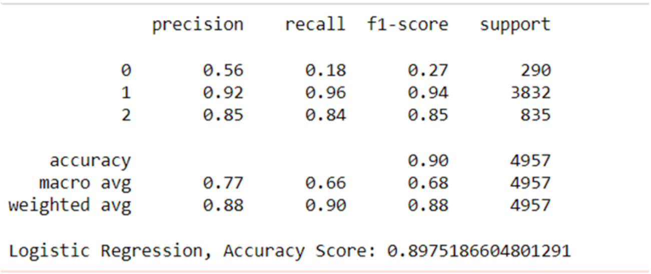
The F1 score is derived by adding together the accuracy and the number of correct answers. Both of these quantities are rates, therefore the harmonic mean seems like the most appropriate choice. As an example of the F1 score formula, consider the following:

With this information, we can get the following equation for the F1 score:

$$\text{F1 score}= 2 * \frac{Precision * Recall}{Precision + Recall}$$

Due to its composition as a mean of Precision and Recall, the F1 score attributes equal importance to the two metrics.

Low F1 scores are the result of poor model performance across both Precision and Recall.In the middle of the F1 scale is where you'll find models with poor Precision but excellent Recall.



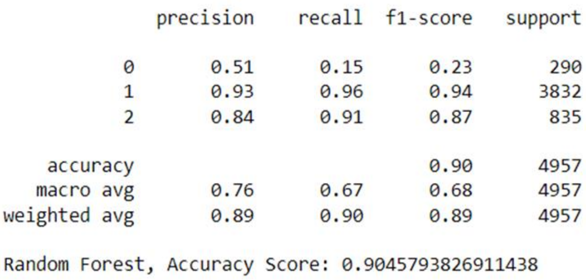
```
              precision    recall  f1-score   support

           0       0.56      0.18      0.27       290
           1       0.92      0.96      0.94      3832
           2       0.85      0.84      0.85       835

    accuracy                           0.90      4957
   macro avg       0.77      0.66      0.68      4957
weighted avg       0.88      0.90      0.88      4957

Logistic Regression, Accuracy Score: 0.8975186604801291
```

Fig 5.6: Accuracy of Logistic Regression



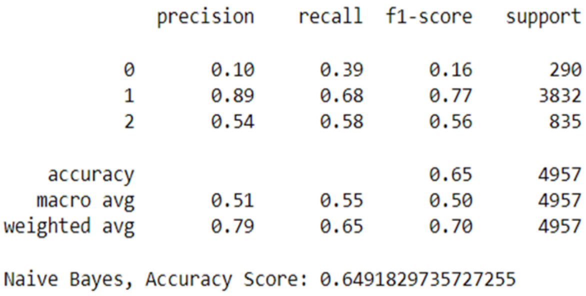
```
              precision    recall  f1-score   support

           0       0.51      0.15      0.23       290
           1       0.93      0.96      0.94      3832
           2       0.84      0.91      0.87       835

    accuracy                           0.90      4957
   macro avg       0.76      0.67      0.68      4957
weighted avg       0.89      0.90      0.89      4957

Random Forest, Accuracy Score: 0.9045793826911438
```

Fig 5.7: Accuracy of Random Forest



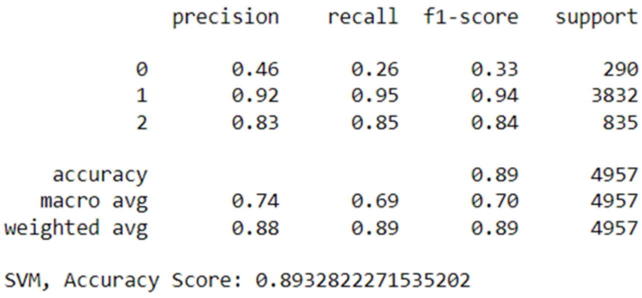
```
              precision    recall  f1-score   support

           0       0.10      0.39      0.16       290
           1       0.89      0.68      0.77      3832
           2       0.54      0.58      0.56       835

    accuracy                           0.65      4957
   macro avg       0.51      0.55      0.50      4957
weighted avg       0.79      0.65      0.70      4957

Naive Bayes, Accuracy Score: 0.6491829735727255
```

Fig 5.8: Accuracy of Naive Bayes



```
              precision    recall  f1-score   support

           0       0.46      0.26      0.33       290
           1       0.92      0.95      0.94      3832
           2       0.83      0.85      0.84       835

    accuracy                           0.89      4957
   macro avg       0.74      0.69      0.70      4957
weighted avg       0.88      0.89      0.89      4957

SVM, Accuracy Score: 0.8932822271535202
```

Fig 5.10: Accuracy of SVM

CONCLUSION:

We started by collecting information to build a dataset on hate speech, which was difficult because what one person considers hate speech may also be deemed everyday language by another. To clean up the dataset and remove any extraneous information, we employ text pre-processing techniques as de-punctuation, tokenizing, stopword removal, stemming, and removing urls and give specifics. The processed text is subsequently subjected to a technique involving the extraction of features includes, but not limited to, doc2vec vector columns, sentiment polarity scores, readability grades, and n-gram tf-idf weights. These traits are then combined into a number of distinct groups so that they can be utilised with a range of categorization methods. The precision and f1-scores of these classification models in regard to various feature sets are evaluated.

The findings show how challenging it is to identify hate speech from other forms of abusive language. It serves as a useful tool for monitoring Twitter for abusive language and highlights the advantages of implementing the recommended improvements. Despite the fact that a thorough investigation of both the features and the errors may result in the development of more reliable methods for the extraction of features and contribute to the solution of existing problems in this industry, it is imperative that such an investigation be carried out.

Even while the number of people using social media continues to rise, the problem of hate speech has not gone away. With this increase comes the necessity for powerful, precise systems capable of detecting this content. Now that we have access to algorithms, we can monitor reviews and comments for instances of hate speech, and this capability will only improve in the future. Eventually, I hope we'll be able to make it such that hate speech online disappears for good.

REFERENCES:

1)T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the issue of abusive language," in Proc. 11th Int. AAA! Con/ Web Social Media, 2017, pp. 512-515.

2)Mr. Z. Waseem "If I'm seeing things, are you racist? Influence of annotations on Twitter's ability to identify hate speech "in Proc. 1st Worh/wp NLP Comput. Social Sci., 2016, pp. 136-143.

3)Z. Zhang, D. Robinson, and J. Tepper, "Detecting hate speech on Twitter using a convolution-GRO based deep neural network," in Proc. Ew: Semantic Web Cont: Heraklion, Greece: Springer, 2018, pp. 741-762.

4)Dillon, W. Benesch, H. Saleem, and D. Ruths, "A Web of Hate: Addressing Hateful Language in Social Network Spaces," 2017, arXiv:1709.10159.

5)E.Wulczyn, N.Thain, and L.Dixon, "Ex machina: Personal abuse viewed at range," in Proc.26th Int.Conf World Wide Web, Apr. 2017,pp.1391-1399.

6)D. Hovy and Z. Waseem, "Hateful images or hateful individuals? predicting characteristics for Twitter offensive speech identification, "2016 NAACL Student Research Workshop Proceedings, pp. 86–94

7)S.Agrawal, M.Shrivastava and S.Sharma, "Degree based classification of damaging speech utilising Data from twitter," arXiv:1806.04197, 2018.

8)Y.Wang and I.Kwok , "Explore the animosity: Designed to detect twitter messages about Blacks," in Proc. 27th AAA/ Co,!f Artif lntel, 2013, pp. 1620-1623.

9)J. Hirschberg and W. Warner, "Capable of recognizing Abuse Speech on the Internet," Pmc. 21ld Workshop on Language and Social Media, June 2012, pp. 17–27.

10)M. L. Wiliams and P. Burnap, "Virtual hateful speech on Twitter: An approach of machine classifying and statistical modelling for policy and decision creation," Policy Internet, vol. 6, no. 3, pp. 221-245, June. 2015.

11)N. Djuric, J. Zhou, R. Morris, V. Radosavljevic, and N. Bhamidipati, "Hate speech identification via comment word embedding," in Proc. 24th int. Conf World Wide Web Companion, 2015, pp. 24–32.

12)T. Joachims, "Getting Massive Supervised Machine Learning Pragmatic," Komplex ittitsreduktion Multi-variate Daten Strukturen, TU Dortmund, Dortmund, Germany, Tech. Rep. 28, 1998.

13)P. Fung and J. Ho Park , "One-step and two-step categorisation for offensive speech identification on Twitter," 2017, arXiv:1705.01207.