Genetic Programming Approach to Detect Hate Speech in social media

NITHIN KUMAR P[1], NARASIMHA M[2,] P NEERAJ ARAVINDHAKSHAN[3], SAI SANJAN REDDY[4], DR. ANITHA.K[5]

[1, 2, 3, 4] *Student, Global Academy of Technology*

[5]*Associate professor, Global Academy of technology*

Abstract - Now-a-days, social media has become part of our day to day lives. More than half of the world makes use of social media. With the increased use of social media, there have been many problems that are rising. Hate speech is one among them. Since there are billions of users using social media, it has become a big problem to deal with as it can be identified based on user's report to such action. It's difficult to monitor and detect the hate speech and on social media platforms. There is a need for detecting and avoiding the hate speech from circulating in social media. The language used and the size of the content pose problem for the traditional machine learning algorithms. Therefore, the genetic programming approach along with ml algorithm is used to detect hate speech because of the better performance it offers.

Indexed Terms—Voting classifier, English Twitter dataset, Universal Sentence Encoder, GP.

I. INTRODUCTION

Around the world, there are almost 4.62 billion users, who use social media in their day to day lives. Social media has become an inseparable part of our lives. Most of the people in the world communicate to each other through social media. The increase in social media usage is due to the facilities they provided for the users to communicate their emotions, feelings and thoughts easily, because these in some situations may lead to beginning and increase of hate regarding others. Though it has many advantages, it has several ill – effects. Hate speech is one among the ill – effects which has changed the concerns for people in social media. Since the number of people moving towards social media is more. The number of individuals inclining towards racism, misogyny has been constituting to violence across the globe. The technology used by the people have also been used by the hate groups who try to organize and circulate hate speech in social media. There is a critical need to track down the hate speech and prevent it from further circulation. Though, there are several measures and steps being taken by the social media sites, which are trying to discover and put a stop to the hate speech in their sites. There are not any effective measures which can be taken to completely curb it, The social media sites mainly depend on the report by the users towards the action. Social media platform depends on the user report, staff and artificial intelligence combination to track and prevent hate speech circulation. Due to Enormous amount of data, it has been a big problem for the platform to track hate speech efficiently. The traditional machine approaches have difficulty due to the language and content used. However, there are some ensemble model and deep learning models which has been used as well for hate speech detection. The genetic programming approach helps in classification of the tweets in an efficient manner. The main purpose of the hate speech detection system is to:

- Design a model to perform classification of the tweets to hate or not hate.
- Use of datasets which are publicly available, to train the model.
- Extracting the feature and perform classification.
- Comparing and displaying the more accurate results by various algorithm.

II. RELATED WORKS

The are several works carried on detection of hate speech in social media. Each study has its own set of advantages and disadvantages. The further study has to take into consideration the drawbacks in previous study. The study done by R. Cao and co., shows that model built has better accuracy but its main drawback is time to build and train LSTM model which is more and the research is based on deep hate which is a deep learning framework [1]. Likewise, the study carried on

Hate speech detection in Indonesian language or Indonesian Twitter made use of many algorithms but with the use of combination of RDFT and LP method gave better accuracy and the model was able to detect the hate speech in twitter of Indonesian language but unable to detect in any other language. The drawback of hate speech detection in Indonesian language is the language, as it can be only used with the Indonesian language [2]. The hate speech detection in Indonesian language made use of tweets available online to classify the dataset into hate or not hate classes. Language was the barrier to the system. It has better accuracy. The model is trained using available public datasets [3]. T. Davidson proposed a study on automatic detection of hate speech. The automated hate speech detection showed higher accuracy since it was an automated classification which was able to differentiate between the classes and provide better accuracy. The drawback was that the classified data was not accurate and there were several misclassifications of data which does not contain any offensive words or any curse terms [4]. Then, several other studies were also carried, which shows the varying range of accuracy. All the points of the previous studies have been taken into consideration for further studies of the hate speech detection system to improvise from the previous studies.

III. SYSTEM DESIGN

System design is an important phase in software development process. The motive is to plan a solution as per requirements specified. It is the initial step for obtaining solution to a problem. The design plays a critical role or acts as a crucial factor which affects the quality of the software. The main objective of the designing phase is output complete design of the software. The aim is to check which model satisfies the system requirements in an efficient way. The user inputs data to system where the data undergoes pre-processing, feature extraction, and finally, classifying the tweets into hate or non-hate classes.

1.1 SYSTEM ARCHITECTURE

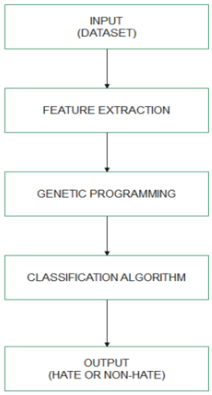

Figure 1.1.1: System Architecture

The model consists of following modules for detecting hate speech:

- Data Acquisition: Extraction and importing of data.
- Data Pre-processing: Cleaning of data and extraction of features.
- Feature Extraction: Vectorizing the text.
- Detection of hate speech: GP approach.
- Model Evaluation: Testing.
- Output: Hate or Not Hate.

The machine learning algorithms like logistic regression, decision tree and voting classifier under genetic programming approach is used for classification of the data.

A. DATA ACQUISITION

The process of collecting the data is called as data acquisition. The dataset used is Twitter data set obtained from Kaggle. It has two columns and 10490 rows. One column lists the tweets and the other column specifies whether the tweet is hate or not hate.

B. DATA PRE-PROCESSING

This step involves cleaning our dataset by removing unnecessary parts of data that would have no role in the prediction task. Cleaning and Organizing of Raw Data to make it suitable for machine learning model
We have performed the following stages of data pre-processing:

Tokenization: break the text present in the tweet into single words.

Remove stop words: removing the common words like this, that etc.
Eliminate punctuation marks: remove \, <,>, /, # etc.

C. FEATURE EXTRACTION

In the present model, the feature is universal sentence encoder extracted using TF-IDF model. It provides better performance and efficiency. It can be applied to combination of sentences and paragraphs. It encodes the text present in the tweet into 512 high - dimensional vectors. These can be used for further classification.

D. HATE SPEECH DETECTION

The aim is to get a classifier which best classifies the tweets into hate or not hate class using genetic programming approach. The module gets the extracted features from the previous module to perform further process. The operators are used to reduce complexity. The genetic programming approach selects the machine learning algorithm with best accuracy for the classification. The performance is evaluated for different dataset. GP model is used for result prediction. The final result gave an accuracy of nearly 85%. The machine learning algorithm like Logistic regression, Decision tree and Majority voting classifier under genetic programming approach are used for classification. The GP model has more accuracy and it requires less information and its capable of providing more optimal solutions to the problem.

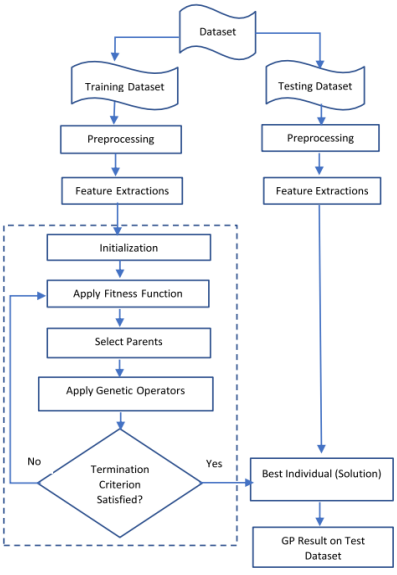

Figure1.1.2: GP model

E. MODEL EVALUATION

To assess the performance of the classifier, the machine learning model can make use of many evaluation measures.

EVALUATION METRICS:

Precision: It is the fraction of all the projected observation which are positive included to positive class.
Precision = True positives/ (True positives + False positives)
Recall: It is the proportion of inspection, predicted to stay in positive category which are in positive category actually. It is used to show how well a model can recognize random positive class observation.
Recall = True Positives/ (True Positives + False Positives)
F1 Score: It is the harmonic mean of recall and precision. It is a mean Evaluation metric.
F1 Score = 2 * Precision * Recall / (Precision + Recall)

1.2 DATA FLOW DIAGRAM

It is used for representing the flow of data through a process or system. It does not have either control or decision loops, rules. It controls input and output entity information.

A. DATA FLOW DIAGRAM LEVEL-0

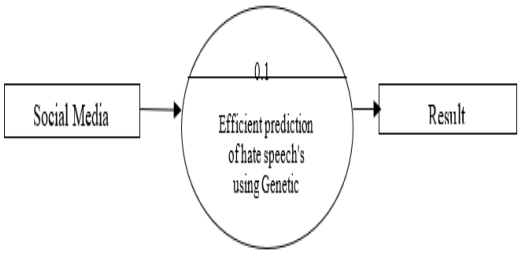

Figure1.2.1: Data Flow diagram Level-0

B. DATA FLOW DIAGRAM LEVEL-1

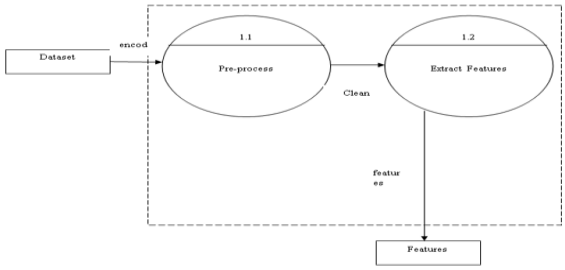

Figure1.2.2: Data Flow diagram Level-1

1.3 USE CASE DIAGRAM

It is dynamic in nature. It is used to provide brief summary about the system details and the users inside the system. It provides graphical representation of interaction among the elements of the system. It consists of elements like actor, use case and the relationship among them.

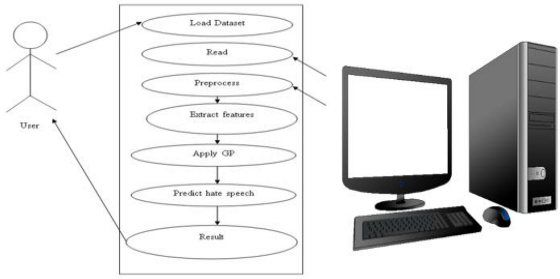

Figure1.3.1: Use case Diagram

1.4 CLASS DIAGRAM

It shows the collection of classes, interface etc. It is also known as the structural diagram. The classes are used to represent the elements, classed that needs to programmed and the interactions within the application.

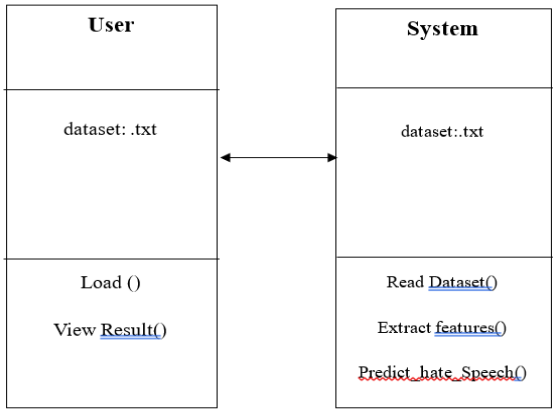

Figure1.4.1: Class Diagram

1.5 ACTIVITY DIAGRAM

It is used to provide description about the changing aspects in the system. It portrays behavior of the system. It models the flow between the activities. In UML, activity diagram is a needful behavioral diagram.

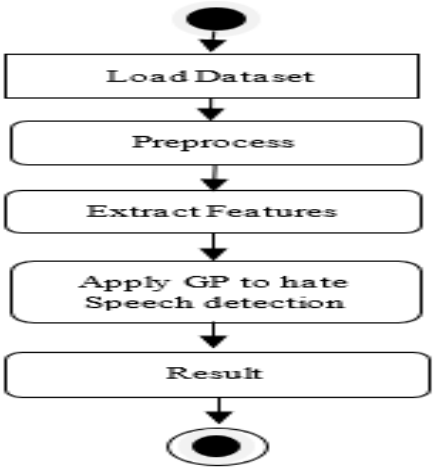

FIGURE 1.5.1: Activity Diagram

IV. RESULTS

The machine learning algorithms like Decision tree, LR (Logistic regression) and Majority voting classifier under the genetic programming approach are used. The classifier with the better accuracy is chosen for classification. The results for various algorithm are shown below.

i. Logistic Regression:

This supervised learning technique has been implemented through sklearn.

The accuracy given by Logstic Regression is 84.70%.

	Precision	Recall	F1 Score	Support
0	0.86	0.94	0.90	1574
1	0.79	0.62	0.69	603

Accuracy			0.85	2177
Macro avg	0.82	0.78	0.79	2177
Weighted avg	0.84	0.85	0.84	2177

Table 1: Shows the values of Evaluation metrics and the accuracy of Logistic Regression algorithm.

ii. Random Forest:

This supervised learning technique has been implemented through sklearn.

The accuracy given by Random forest is 84.60%.

	Precision	Recall	F1 Score	Support
0	0.86	0.94	0.90	1574
1	0.79	0.62	0.69	603

Accuracy			0.85	2177
Macro avg	0.82	0.78	0.79	2177
Weighted avg	0.84	0.85	0.84	2177

Table 2: Shows the values of Evaluation metrics and the accuracy of Random Forest algorithm.

iii. Voting Classifier:

This supervised learning technique has been implemented through sklearn.

The accuracy given by Voting Classifier is 80.18%.

	Precision	Recall	F1 Score	Support
0	0.79	0.99	0.88	1574
1	0.93	0.30	0.46	603

Accuracy			0.80	2177
Macro avg	0.86	0.65	0.67	2177
Weighted avg	0.83	0.80	0.76	2177

Table 3: Shows the values of Evaluation metrics and the accuracy of Voting Classifier.

iv. Comparison of Accuracy of different Algorithms:

Logistic Regression and Random Forest algorithm has maximum accuracy.

Algorithm	Accuracy
Logistic Regression	84.70%
Voting Classifier	80.18%
Random Forest	84.60%

Table 4: Accuracy comparison of algorithms.

V. CONCLUSION

According to the study, we found ml algorithms were efficient enough to classify the tweets to hate or not-hate. There are certain limitations in the current research that are addressed in coming studies. The accuracy may vary because of the size of the dataset that is used. More the dataset size, classification will be more accurate and precise.

The Supervised machine learning algorithms require prior training to perform classification efficiently. The classification mainly depends on the size of the dataset which it has been trained with. With the results, we can be able to classify the tweets into hate or non-hate with the help of genetic programming approach.

VI. REFERENCES

[1] R. Cao, R. K.-W. Lee, and T.-A. Hoang, ‘‘DeepHate: Hate speech detection via multi-faceted text representations,’’ in Proc. 12th ACM Conf. Web Sci., Southampton, U.K., Jul. 2020, pp. 11–20.

[2] M. O. Ibrohim and I. Budi, ‘‘Multi-label hate speech and abusive language detection in Indonesian Twitter,’’ in Proc. 3rd Workshop Abusive Lang. Online, Florence, Italy, Aug. 2019, pp. 46–57.

[3] I. Alfina, R. Mulia, M. I. Fanany, and Y. Ekanata, ‘‘Hate speech detection in the Indonesian language: A dataset and preliminary study,’’ in Proc. Int. Conf. Adv. Comput. Sci. Inf. Syst. (ICACSIS), Jakarta, Indonesia, Oct. 2017, pp. 233–238.

[4] T. Davidson, D. Warmsley, M. Macy, and I. Weber, ‘‘Automated hate speech detection and the problem of offensive language,’’ in Proc. ICWSM, Montreal, QC, Canada, May 2017, pp. 15–18

[5] Z. Waseem and D. Hovy, ‘‘Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter,’’ in Proc. NAACL Student Res. Workshop, San Diego, CA, USA, Jun. 2016, pp. 88–93.

[6] P. Fortuna and S. Nunes, ‘‘A survey on automatic detection of hate speech in text,’’ ACM Comput. Surv., vol. 51, no. 4, pp. 1–30, Sep. 2018.

[7] T. John, ‘‘Hate speech,’’ in Encyclopedia of the American Constitution, L. Levy, K. Karst and A. Winkler, 2nd ed. New York, NY, USA: Macmillan, 2000, pp. 1277–1279.

[8] Z. Al-Makhadmeh and A. Tolba, ‘‘Automatic hate speech detection using killer natural language processing optimizing ensemble deep learning approach,’’ Computing, vol. 102, no. 2, pp. 501–522, Feb. 2020.

[9] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, ‘‘Deep learning for hate speech detection in tweets,’’ in Proc. 26th Int.Conf.World Wide Web Companion (WWW Companion), Perth, WA, Australia, Apr. 2017, pp. 759–760.

[10] T. Putri, ‘‘Analisis dan deteksi hate speech pada sosial Twitter berbahasa Indonesia,’’ M.S. thesis, Dept. Comp. Sci., Indonesia Univ., Indonesia, 2018.

[11] J. Sachdeva, K. K. Chaudhary, H. Madaan, and P. Meel, ‘‘Text based hatespeech analysis,’’ in Proc. Int. Conf. Artif. Intell. Smart Syst. (ICAIS), Coimbatore, India, Mar. 2021, pp. 661–668.

[12] J. Koza, ‘‘Genetic programming: On the programming of computers by means of natural selection,’’ Stat. Comput., vol. 4, no. 2, pp. 87–112, Jun. 1994.

[13] A. Esparcia-Alcázar, A. Ekárt, S. Silva, S. Dignum, and A. Uyar, ‘‘Genetic programming,’’ in Proc. EuroGP, Istanbul, Turkey, Apr. 2010, pp. 7–9.

[14] C.-S. Kuo, T.-P. Hong, and C.-L. Chen, ‘‘Applying genetic programming technique in classification trees,’’ Soft Comput., vol. 11, no. 12, pp. 1165–1172, Aug. 2007.