Hate Speech Classification Implementing NLP and CNN with Machine Learning Algorithm Through Interpretable Explainable AI

Mahmudul Hasan Shakil
Dept. of Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
mahmudul.hasan.shakil@g.bracu.ac.bd

Md. Golam Rabiul Alam
Dept. of Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
rabiul.alam@bracu.ac.bd

***Abstract*—In recent years, data innovation has progressed quickly, and online media has gone through a problematic transformation. Destinations like Facebook, Twitter, and Instagram where clients can communicate their thoughts through messages, photographs, and recordings have become famous. Sadly, be that as it may, it has turned into a favorable place for disdain discourse, affronts, cyberbullying, and mysterious dangers. There are many examinations around here, however, none have given an adequate degree of exactness. This article proposes a Convolutional Neural Network (CNN) and Natural Language Processing (NLP) amalgamation strategy that characterizes malicious and non-malicious remarks at a beginning phase and groups them into six classifications utilizing Wikipedia's talk page edits gathered by Kaggle. The proposed engineering changes over words into words by utilizing NLP strategies like tokenization and stemming, and word embedding as well as data preprocessing methods that transform words into vectors. In the subsequent advance, the handled informational index gathered in the initial step is applied through 5 distinct classifiers (XGBoost, Random Forest, Decision Tree, AdaBoost, and Gradient Boosting classifier) and an explainable artificial intelligence (XAI) strategy (LIME). In the last stage, all classifiers were joined to accomplish a malicious comment classification accuracy of 99.75, which is higher than the past work.**

Keywords—NLP; CNN; XGBoost; Random Forest; Decision Tree; AdaBoost; Gradient Boosting; Explainable AI; LIME.

I. INTRODUCTION

As a result of the enhancements of innovation, cyberbullying or online provocation utilizing oppressive, foul, or awful language has become extremely simple for this age. As a result of its compass and availability, online media has advanced into a center for data and area-related look. It has given every individual the inspiration to join themselves before others. Sadly, this stage is additionally turning into a stage for detesting and scrutinizing individuals who advance variety in shading, identity, orientation, and sexual direction, in any event, putting them at risk for viciousness. Cyberbullying and provocation have advanced into genuine worries that influence a wide range of clients, at times bringing about major mental issues like gloom or even self-destruction.

The objective of this examination is to guarantee that web media is liberated from any hurtful substance and comments. The motivation behind this study was to check whether there were any hostile remarks via web-based media networks utilizing profound learning. furthermore, to additionally sort them into harmful, serious poisonous, indecent, affront, danger, and character can't stand classes. We likewise endeavored to evaluate the adequacy of every calculation's dataset. We are utilizing Natural Language Processing (NLP) and Convolutional Neural Network (CNN) in the main stage. Data preprocessing techniques as well as NLP techniques like tokenization and stemming, and word embedding techniques to turn words into vectors, are used to arrange the proposed architecture. We take the first phase's processed data set and run it through five distinct classifiers in the second phase. We measure the accuracy of these classifiers and get 99.72% for XGBoost, 99.78% for Random Forest, 99.60% for Decision Trees, 99.68% for Adaboost, and 99.69% for Gradient Boosting classifiers. We also introduce these classifiers with an Explainable Artificial Intelligence (XAI) method to see the comparison between actual values and predicted values. We choose the LIME model for the XAI method. We ensemble all the classifiers and evaluate several metrics in the final phase. For toxic comment classification, we measure Accuracy: 99.76%, Precision: 99.40%, Recall: 99.35%, and F1 score: 99.38%, which is higher than existing works.

The sections of the paper are organized as follows to represent the research work: The related work in this area is discussed in Section 2, and the proposed methodology is outlined in Section 3. Section 4 presents the experimental analysis and the procedure's subsequent implementation and findings, and Section 5 summarizes the study work's objective and future development opportunities.

II. LITERATURE REVIEW

Online maltreatment and the utilization of unsafe language have turned into an issue as web-based media has filled in prevalence as of late. There isn't a great deal of work done on this point to fix the issue. Here is a portion of the papers we're checking out:

In this paper [1], the robotized extraction of information from qualifying models will prompt a leap forward in the effective utilization of information for quiet pursuit in clinical data sets. An extensive piece of capability models consolidates vaporous data about diseases and occasions. This organization fosters a progressive Natural Language Processing (NLP) pipeline for separating and characterizing transient information as noteworthy, current, and coordinated by utilizing free-text qualifying models. This paper [2] portrays the three subjects: since most computer-based intelligence calculations don't give explanations to conjectures, building consistent systems is a key theme in the field of Natural Language Processing (NLP). Generally, existing methodologies for reasonable computer-based intelligence structures will zero in on unraveling the yields or the connections between information sources and yields. Despite this, fine-grained information is as often as possible overlooked, and systems don't continuously give justifiable clarifications.

The convolutional neural network (CNN) is a kind of brain network that has been generally used to address order issues in an assortment of spaces, including text arrangement. We found a couple of papers on this point while leading our examination. One of those interesting examinations was exclusively committed to message grouping utilizing CNN. Kim et al [3] showed a progression of sentence-level arrangements that were performed utilizing CNN. The creators exhibited how a modest quantity of change and a static vector can deliver recognizable outcomes on an assortment of benchmarks. Task-explicit and static vectors are utilized in their proposed worldview. The significance of pre-prepared and solo word vectors in NLP was additionally exhibited by the creators. Georgakopoulos and his partners S. V. et al [4] utilized the CNN model to take care of a destructive remark characterization issue in another review. For more predictable investigation, the fundamental dataset was transformed into a subset, which was then used for paired groupings to sift through perilous comments. Zhang et al. [5] presented another technique that consolidated convolutional brain organizations (CNN) and gated repetitive organizations (GRU) and was found to upgrade order precision experimentally. The writers accumulated information from openly accessible Twitter datasets and fostered another one by incorporating tweets about displaced people and Muslims that were media-centered at the hour of composing attributable to different late events. The creators utilized a little pre-handling on tweets prior to involving CNN+GRU engineering in layers that included word installing, 1D convolutional layer, 1D max-pooling layer, GRU, and SoftMax layer. They led a relative investigation of the datasets and showed that their proposed strategy beat baselines and created more prevalent outcomes than the other datasets revealed discoveries. It lays out the current standard by scoring 1 to 13% in F1 on six out of seven datasets.

The creators of this paper [6] utilized XGBoost to identify counterfeit news, which is basically indistinguishable from what we're doing. The scientists used consideration-based models that main used text information. The creators of another paper [7] distinguish and forestall online media animosity. They've utilized XGBoost, Support Vector Machine (SVM), and Gradient Boosting Classifier, among other AI calculations for grouping (GBM) The datasets in Hindi, English, and Hindi-English (code-mixed) were examined. The issue is generally appropriate for the support vector machine (SVM) and the hereditary calculation (GBM). To identify online maltreatment, Wikipedia utilizes a human-driven approach. The creators [8] give a worldview to understanding and identifying such abuse in the English Wikipedia people group. They depend on Wikipedia information sources that are available to people in general.

III. Methodology

The main period of the proposed approach's work process is pictured in Figure. 1 flowchart shows that after bringing in the dataset and parting it into preparing and testing sets, the work process is shown. Following that, the dataset goes through a preprocessing stage that incorporates information purging, tokenization, stemming, and word implanting. We tried three notable word inserting procedures and observed that the outfit in addition to the CNN classifier was the most reliable. The CNN design utilizes parallel grouping to recognize regardless of whether remarks are poisonous and afterward predicts subclass poisonous levels assuming that the remarks are destructive.

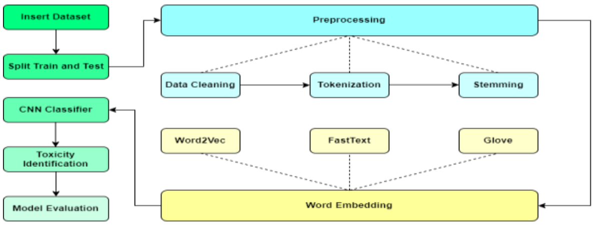

Fig. 1. The first phase of the workflow

A. Data Preprocessing:

1) Data Cleaning: Data cleaning is fundamental for producing better outcomes and quicker handling by eliminating abnormality from the dataset. Stop words are taken out [9], accentuations are taken out, all words are changed to bring down the case, copy words are taken out, URLs, emoticons, or shortcodes of emoticons are eliminated, numerals are taken out, one person based word is taken out, and images are taken out. Normal Language Tool stash (NLTK) [10] is a python library for an assortment of dialects that we utilized in our model to further develop classification and precision.

2) Tokenization: It is the most common and vital part of NLP, where a sentence brimming with words is isolated or broken into individual words [11], every one of which is alluded to as a token. For making an interpretation of the word to a vector number, a model called FastText is used.

3) Stemming: Stemming is a method for planning words by erasing or bringing down the emphasis types of words like playing, played, and perky to find the root, otherwise called a lemma. There are additions, prefixes, tenses, sexual orientations, and other linguistic structures in these words. Moreover, assuming we inspect a gathering of words and observe a root that isn't of a similar sort, we think of it as a different class of that word, known as a lemma. For a better result, we apply the lemma approach in our model [12].

B. *Word Embedding:*

The portrayal of a vector constructed utilizing brain networks is learned through word installation. It's for the most part used to control word vector portrayals in a significant other option. Word implanting changes vector portrayals for mathematical words by making an interpretation of semantic information to an inserted space [13].

C. *CNN Architecture for Classification:*

Considering its innate ability to utilize two factual characteristics known as 'local stationarity' and 'compositional structure,' Convolutional Neural Network, or CNN, has been broadly used to tackle picture order [14] issues. The main rule is to execute CNN for harmful remark order [15], sentences should be encoded prior to being taken care of [16] to CNN engineering. To work on the situation, the methodology of involving jargon in a media of record containing words, which has sets of texts planned into number lengths going from 0 to 1, was utilized. From that point forward, the cushioning approach is utilized to fill the archive network with zeros to accomplish the most extreme length, as CNN engineering requires contact input dimensionality. The subsequent stage is to change over the encoded reports into networks, with each column compared to a solitary term. The inserting layer, in which a thick vector changes any word (column) into a portrayal of low aspects [17], moves the frameworks created. Our CNN configuration is contained a 50 unit completely connected (thick) layer with a part size of five out of 128 channels for five-word embeddings. The inserting strategy utilizes fixed thick word vectors created with programs like FastText, word2vec, and GloVe, which were talked about in the past segment. We utilize the ADAM analyzer and twofold cross-entropy misfortune to prepare our model, which we assess with paired precision in the principal stage prior to continuing to multi-class arrangement for a dangerous evening out. We utilize four ages for high register power, skewering the preparation informational index into small clusters of 64 examples, with 70% of the information being utilized for preparing and 30% for testing.

D. *Applying Classifiers:*

After getting the processed data set that is collected through the CNN Classifiers, we deploy this on five different classifiers. We have used Extreme Gradient Boosting (XGBoost) classifier, Random Forest classifier, Decision Trees classifier, AdaBoost classifier, Gradient Boosting classifier. We dropped the one-of-a-kind id esteem and took just the worth of six classes. Then, at that point, we split the informational index into train and test where 60% of the information is utilized for preparing and 40% for testing. Furthermore, we take the worth of an irregular state as 7. From that point forward, we change the informational collection into the test and train the informational index that is prepared to apply through classifiers. Figure 2 is showing

the second period of our exploration work.

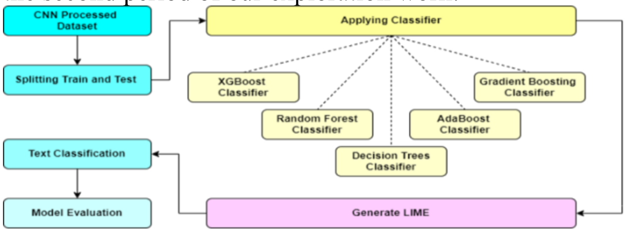

Fig. 2. The second phase of the workflow

E. *Generate Explainable AI:*

Artificial Intelligence (AI), a huge topic, has exploded in popularity in recent years. AI models have begun to surpass human intelligence at a rate no one could have imagined, as more and more complex models are released each year. Explainable AI refers to a set of approaches or strategies for explaining the decision-making process of a particular AI model [18]. With more and more advanced strategies emerging each year, this relatively new branch of AI has shown immense promise. We used one Explainable AI (XAI) method for the result evaluation of these classifiers. We choose the LIME model for applying the XAI method. Local Interpretable Model-Agnostic Explanations (LIME) is an acronym for Local Interpretable Model-Agnostic Explanations [19]. We take the positive comment as well as a negative comment that was previously trained through all five classifiers. And apply those comments through the LIME model and get different results.

F. *Appending Classifiers:*

In the last stage, we affix this large number of classifiers into one rundown and count the F1-weighted score for every classifier. Then, at that point, we utilize K-fold cross-validation where n split is 10 and random state is 12. Cross-validation is a resampling strategy for assessing AI models on a little example of information. That is, little example size is utilized to assess how the model would act overall when it is utilized to create forecasts on information that was not utilized during preparation. It's a well-known technique since it's not difficult to handle and creates a less slanted or hopeful gauge of model skill than different methodologies, like a basic train/test split.

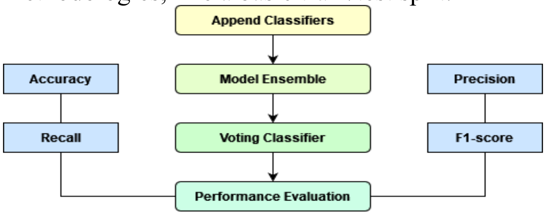

Fig. 3. The final phase of the workflow

G. *Ensemble Model:*

From that point onward, we present outfit realization which is displayed in Figure 3. Gathering learning is a far-reaching nonexclusive method for AI that joins the forecasts from various models to work on prescient execution. Group approaches are models that are made in products and afterward consolidated to come by better outcomes. Much of the time, gathering approaches give more exact outcomes than a solitary model. In various AI rivalries, the triumphant arrangements utilized troupe draws near.

We utilized a Voting Classifier to outfit every one of the classifiers. A Voting Classifier is an AI model that gains from an outfit of many models and predicts a result (class) in view of the greatest likelihood of the result being the picked class. Rather than developing separate devoted models and deciding their precision, we make a solitary model that is prepared by various models and predicts yield in view of their total larger part of decisions in favor of each result class. We utilize delicate deciding in favor of our outfit model and get various measurements of exactness, review, f1 score, and accuracy.

IV. Result and Analysis

First, we'll go over the Kaggle dataset we utilized and how we visualized the categories and their relationships in this part. Following that, the suggested system's performance on this dataset for harmful comment classification is demonstrated. Finally, a demonstration of the overall process is shown.

A. *Dataset:*

The dataset we used for this study came from Kaggle, and it's called "Wikipedia Talk Page Comments tagged with toxicity causes" [20], and it contains about 1,60,000 comments with manual labeling. There are six classes in the dataset (toxic, severe toxic, obscene, insult, threat, and identity hate). The correlation matrices in Figure 4 demonstrate that "toxic" remarks are most significantly associated with the "insult" and "obscene" classes. The only weak correlation is between 'toxic' and 'threat.' 'obscene' and 'insult' comments are also highly connected, which makes sense. It also reveals that the class 'threat' has the smallest association with any of the other classifications.

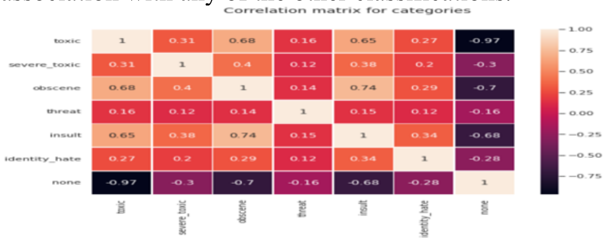

Fig. 4. Visual representation of the correlation between classes

B. *CNN Classification Evaluation:*

After preprocessing with tokenization and stemming, we employed a CNN model using the FastText embedding approach for binary classification in the early stages. Convolutional structures are used in three different ways. At the same time, three different convolution structures are used, each having a dense vector dimension of 300 and a filter width of 128. The filter width was equal to the vector dimension for increasing convolutional layer height, and it was 3, 4, and 5 for increasing convolutional layer height. Following each convolutional layer, there is a cumulative pooling process. The output of the pooling layer is a whole layer attached, whereas the SoftMax feature relates to the terminating layer. For each epoch, Figure 5 shows the training and testing loss.

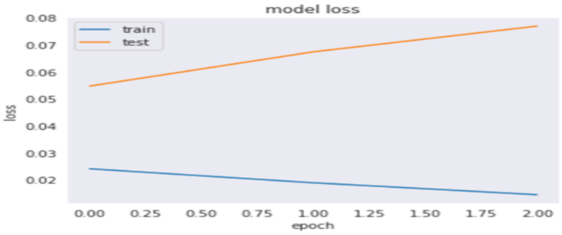

Fig. 5. Loss function on each epoch for train and test set

Table 1 depicts a toxic leveling demonstration using some random nasty and toxic comments, with predicted toxicity depending on the six classifications. It predictably divides each poisonous term into sub-classes, with some sentences falling into multiple categories or scoring exceptionally well in a single category.

TABLE I. Toxic comment labeling

Comment	Toxic %	Severe Toxic%	Obscene %	Threat %	Insult %	Identity Hate%
“Go jump off a bridge jerk”	98	5	91	1	87	0
“I will kill you”	100	2	9	91	6	0
“Have a nice day”	0	0	0	0	0	0
“u r ugly as my jackass”	100	4	97	0	98	0
“you son of a bitch”	100	13	100	0	100	0
“Stupid boy!!”	100	0	24	0	97	0
“Let me fuck you”	100	12	100	0	86	0
“you bastard!”	100	1	100	0	99	0
“I am a Human”	1	0	0	0	0	0

C. *Classifiers Evaluation:*

After applying CNN classifier to the dataset, we got a processed dataset. Then, we apply it through five classifiers and get accuracy, precision, recall, and f1 score for each classifier. We also tried to show the learning curve, validation curve, feature importance, and recursive feature elimination with cross-validation. Figures 6, 7, 8, 9, and 10 are showing the learning curve of each classifier.

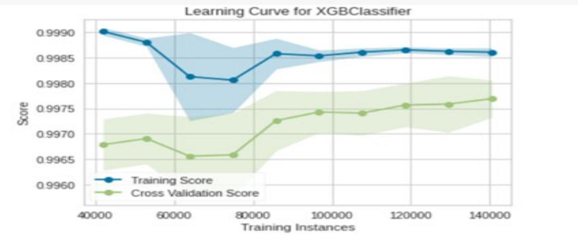

Fig. 6. Learning Curve for XGBoost

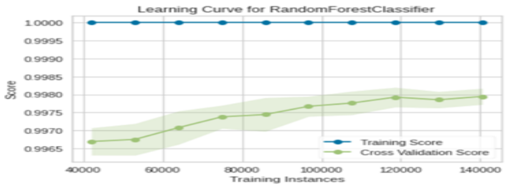

Fig. 7. Learning Curve for Random Forest

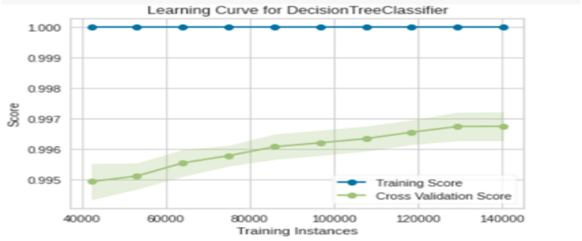

Fig. 8. Learning Curve for Decision Tree

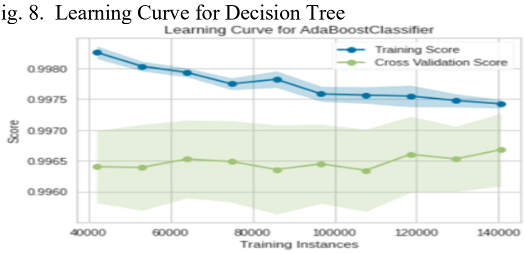

Fig. 9. Learning Curve for AdaBoost

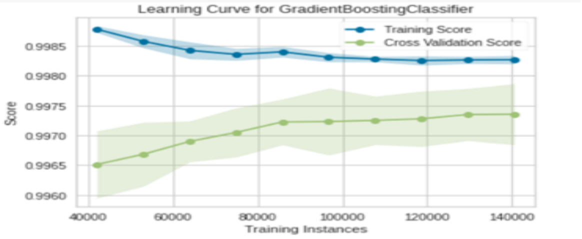

Fig. 10. Learning Curve for Gradient Boosting

On the other hand, recursive feature elimination with cross-validation adds cross-validation to the mix. The validation data is used to calculate the score for feature importance. Depending on the quantity of the data and the estimator employed, this can be a resource-intensive procedure. The recursive feature elimination with cross-validation (RFECV) for each classifier is depicted in Figures 11, 12, 13, 14, and 15.

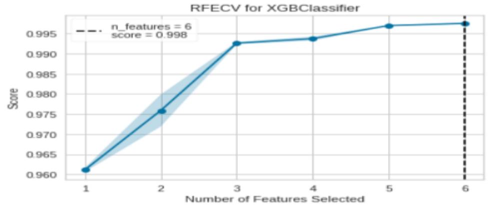

Fig. 11. RFECV for XGBoost

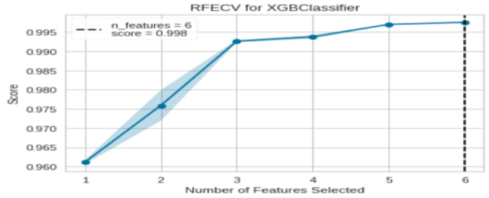

Fig. 12. RFECV for Random Forest

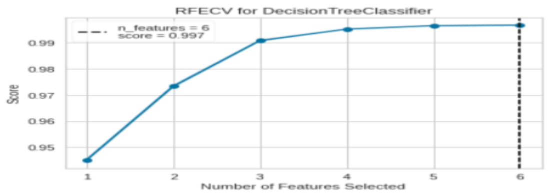

Fig. 13. RFECV for Decision Tree

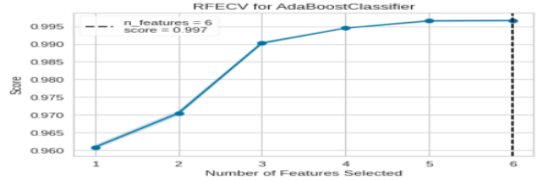

Fig. 14. RFECV for AdaBoost

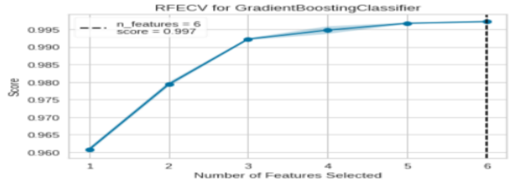

Fig. 15. RFECV for Gradient Boosting

In the final phase, all the five classifiers are appended to an estimator. We applied f1-weighted as scoring in cross-validation score for model selection. We deployed Kfold as cv where the no of splits is 10 and the random state is 12.

D. *Ensemble model:*

Then, we ensemble all the classifiers into one. We used a voting classifier where voting = soft. Figure 16 is showing the classification report for the ensemble model.

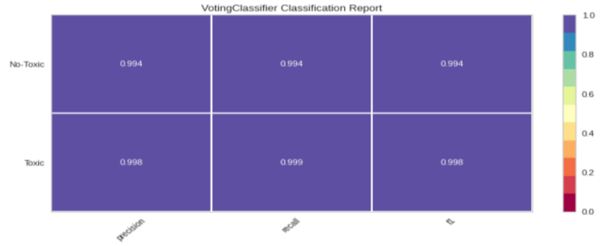

Fig. 16. Classification report

From the classification report, we measure some important metrics of our ensemble model. Table 2 is showing the metrics evaluation result for the ensemble model.

TABLE II. METRICS EVALUATION FOR ENSEMBLE MODEL

Metrics	**Score**
Accuracy	99.75%
Precision	99.39%
Recall	99.34%
F1 score	99.37%

After that, we demonstrate a cross-validation report for the voting classifier where we calculate the mean score for this method is 0.997. The cross-validation score for the voting classifier is shown in figure 17.

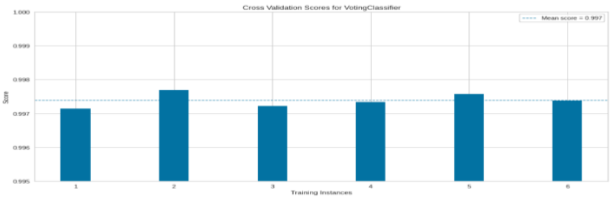

Fig. 17. Cross-validation score for Voting Classifier

At last, we generate the LIME model as the XAI method in our ensemble model. We randomly select one toxic comment and one non-toxic comment and apply them through our ensemble model to generate the LIME explanation.

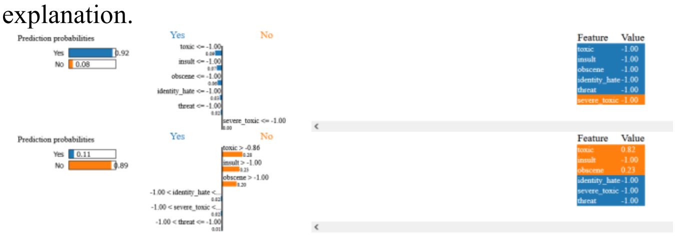

Fig. 18. LIME explanation of ensemble model

Figure 18 is showing the LIME explanation for our ensemble model. For the toxic comment, it takes toxic, insult, obscene, identity hate, and threat values. But for the non-toxic comment, it takes only toxic, insult, and obscene class values for high intensity. This model can predict the toxic comment with a “Yes” percentage rate of 92% and detect the non-toxic comment with a “No” percentage rate of 89% which is very much efficient as compared to other existing works.

V. CONCLUSION

In this work, we give a harmful remark characterization framework, which is a significant theme since, with the ascent of online media, it is likewise urgent to keep away from cyberbullying, foul, or noxious remarks, which is as yet hard to forestall. In any case, by joining CNN with a quick text word implanting procedure after regular language handling, including information cleaning, tokenization, and stemming, we had the option to accomplish more noteworthy exactness contrasted with other current works. We applied the CNN handled dataset to XGBoost, Random Forest, Decision Tree, AdaBoost, and Gradient Boosting Classifiers and create a LIME clarification of these calculations. In the last stage, we produce every one of the classifiers into one and group them. A voting classifier is applied to our troupe model to track down various measurements and assess the general works.

We attempted to convey different calculations in contrast with other existing works which make our works nobler. It is added to our greatest advantage to involve the framework in web-based media and instructive stage visits enclose the not-so-distant future since these two stages are inclined to get huge volumes of antagonism and harmful remarks. We additionally need to anticipate dealing with voice information acknowledgment.

REFERENCES

[1] M. Anand and R. Eswari, "Classification of Abusive Comments in Social Media using Deep Learning," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), 2019, pp. 974-977, doi: 10.1109/ICCMC.2019.8819734.

[2] M. Ibrahim, M. Torki and N. El-Makky, "Imbalanced Toxic Comments Classification Using Data Augmentation and Deep Learning," 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 2018, pp. 875-878, doi: 10.1109/ICMLA.2018.00141.

[3] Van Hee, C., Jacobs, G., Emmery, C., Desmet, B., Lefever, E., Verhoeven, B., ... & Hoste, V. (2018). Automatic detection of cyberbullying in social media text. PloS one, 13(10), e0203794.

[4] Georgakopoulos, S. V., Tasoulis, S. K., Vrahatis, A. G., & Plagianakos, V. P. (2018, July). Convolutional neural networks for toxic comment classification. In Proceedings of the 10th Hellenic Conference on Artificial Intelligence (pp. 1-6)

[5] Saeed, H. H., Shahzad, K., & Kamiran, F. (2018, November). Overlapping toxic sentiment classification using deep neural architectures. In 2018 IEEE International Conference on Data Mining Workshops (ICDMW) (pp. 1361-1366).

[6] Srivastava, S., Khurana, P., & Tewari, V. (2018, August). Identifying aggression and toxicity in comments using capsule network. In Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018) (pp. 98-105).

[7] Kandasamy, K., & Koroth, P. (2014, March). An integrated approach to spam classification on Twitter using URL analysis, natural language processing, and machine learning techniques. In 2014 IEEE Students' Conference on Electrical, Electronics and Computer Science (pp. 1-5). IEEE.

[8] Anand, M., & Eswari, R. (2019, March). Classification of Abusive Comments in Social Media using Deep Learning. In 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) (pp. 974-977)

[9] Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. Information Processing & Management, 50(1), 104-112.

[10] Hardeniya, N., Perkins, J., Chopra, D., Joshi, N., & Mathur, I. (2016). Natural language processing: Python and NLTK. Packt Publishing Ltd.

[11] Orbay, A., & Akarun, L. (2020). Neural sign language translation by learning tokenization. arXiv preprint arXiv:2002.00479.

[12] Hidayatullah, A. F., Ratnasari, C. I., & Wisnugroho, S. (2016). Analysis of Stemming Influence on Indonesian Tweet Classification. Telkomnika, 14(2), 665.

[13] Yang, X., Macdonald, C., & Ounis, I. (2018). Using word embeddings in Twitter election classification. Information Retrieval Journal, 21(2-3), 183-207

[14] M. I. Pavel, A. Akther, I. Chowdhury, S. A. Shuhin and J. Tajrin. (2019). Detection and recognition of Bangladeshi fishes using surf and convolutional neural network. Int. J. of Adv. Res. 7 (Jun). 888-899

[15] Risch, J., & Krestel, R. (2020). Toxic Comment Detection in Online Discussions. In Deep Learning-Based Approaches for Sentiment Analysis (pp. 85-109).

[16] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. Journal of machine learning research, 12(ARTICLE), 2493-2537.

[17] Gal, Y., & Ghahramani, Z. (2016). A theoretically grounded application of dropout in recurrent neural networks. In Advances in neural information processing systems (pp.1019-1027)

[18] C. J. Mahoney, J. Zhang, N. Huber-Fliflet, P. Gronvall and H. Zhao, "A Framework for Explainable Text Classification in Legal Document Review," 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 1858-1867, doi: 10.1109/BigData47090.2019.9005659.

[19] S. Gupta and G. Sikka, "Explaining HCV prediction using LIME model," 2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC), 2021, pp. 227-231, doi: 10.1109/ICSCCC51823.2021.9478092.

[20] www.kaggle.com/c/jigsaw-toxic-commentclassification-challenge/data