Educational Administration: Theory and Practice
2024, 30(5), 918-926
ISSN: 2148-2403
https://kuey.net/

Research Article

Prediction Of Hate Speech Classification Using Secured Supervised Machine Learning With Nlp

Dr.Kavitha Subramani [1]*, Hemalatha.P [2] ,Dr.M.S.Vinmathi[3], Kavya.S [4], Aswini.T[5]

[1] * Professor, Department of CSE, Panimalar Engineering College, Chennai, India.
[2]UG Scholar, Department of CSE, Panimalar Engineering College, Chennai, India.
[3]Professor, Department of CSE, Panimalar Engineering College, Chennai, India.
[4]UG Scholar, Department of CSE, Panimalar Engineering College, Chennai, India.
[5]UG Scholar, Department of CSE, Panimalar Engineering College, Chennai, India.
*corresponding author email: kavitha.pec2022@gmail.com

Citation: Dr.Kavitha Subramani et.al , (2024), Prediction Of Hate Speech Classification Using Secured Supervised Machine Learning With Nlp , *Educational Administration: Theory And Practice*, *30*(5), 918-926
Doi: 10.53555/kuey.v30i5.2988

ARTICLE INFO | **ABSTRACT**

The daily lives of many individuals have been profoundly impacted by digital media. Hate speech refers to a story that is meant to distract or deceive the audience. Due to a number of factors, including the rise of online social networks during the recent years, hate speech has increased in frequency in the online world. Online social network users can easily be impacted by this hatred speech. Hate speech has become an issue for society, sometimes spreading more quickly than the truth. All of this hate speech is undetectable to a person. Therefore, a machine learning algorithm that can recognize hate speech automatically is required. Machine learning models are developed using algorithms to determine whether a speech is hate speech, hurtful speech, or neither. When compared to the other algorithms, the Gradient Boosting Algorithm produces the greatest accuracy. Therefore, the Gradient Boosting Algorithm is used for project launch. The Kaggle dataset used for this project to classify hate speech contains features like count, hate speech, offensive, neither hate nor offensive, class, and tweet.

Keywords: Hate speech, Offensive speech, Machine learning, Social networks, Online , Algorithm

1.Introduction

Hate speech is a pernicious form of discourse that can spread quickly on social media, as a result of biases, or as a result of conflicts between various groups both domestically and internationally. Hate crimes are crimes committed against a person due to their actual or perceived membership in a specific group. Hate speech is defined by Facebook's protected characteristics as an assault on a person's dignity, including their race, ethnicity, or place of birth. According to Twitter's terms of service, users are prohibited from threatening or harassing others via tweets according to their race, gender, religion, or any other trait. YouTube censors content that incites aggression or hatred against particular people or groups in addition to content that is restricted based on age, caste, and handicap. Hate speech related to internet radicalization or criminal activity is frequently researched.

2.Related Work

Making use of deep learning algorithms [1] created a multi-domain hate speech corpus (MHC) of English tweets, which contains hate speech against religion, nationality, ethnicity, and gender in general and covers a variety of topics, including politics, terrorism, technology, natural disasters, and human/drug trafficking. Each occurrence in our dataset is carefully classified as either containing hate or not. To detect hate speech from Twitter data, we employ the most recent state-of-the-art models and present a stacked-ensemble-based hate speech classifier (SEHC). Our findings suggest that the suggested approach might provide a reliable starting point for subsequent research using this dataset. [2] Hate speech detection uses a range of recurrent neural frameworks also known as RNNs, for categorization, for instance the one known as the Gated Recurrent Unit (GRU). The most effective experimental outcomes were those for hatred speech detection, Word2Vec embedding, and RNN-GRU.[3] the first benchmark collection on hate speech that addresses various facets of the problem. Each post in the set of data has been documented in several different angles: the intended community, the basic, widely-used 3-class classification, and the rationales. Models that take into account human reasoning lessen bias against specific populations. [4] Convolutional and recurrent layers were combined in an extensive natural language processing (NLP) model for the automatic identification of discriminatory language in online social network statistics. This model was applied to the HASOC2019 collection and attained an aggregate F1 value about 0.63 in hatred recognition of speech on the HASOC test set. [5] explains several distinct Deep Neural Network, or DNN, Frameworks for Twitter hate speech detection, including the one known as the Gated Recurrent Unit (GRU), which is good at capturing sequence orders, the Convolution Neural Network, or (CNN), which is good at feature extraction. WikiText103 dataset was used for pre-training the AWD -LSTM model. This approach fared notably better than the other Architectures. 6] compared the performance of eight machine learning algorithms and three feature engineering approaches to assess how well they performed on a publicly available dataset with three different classes. The results of various comparisons will be used to compare current studies for extant automated text classification techniques in the future. [7] aimed to look into a number of neural network models built on recurrent and convolutional neural networks to identify hate speech in Arabic tweets. The CNN model performs best, with an F1-score of 0.79 and an AUROC of 0.89..[8] Different machine learning algorithms based on classification and regression models are used to teach the computer using various datasets. Twitter messages with the categories "offensive" and "not offensive" make up the datasets. Tamil language model received an F1 value of 0.87 while Malayalam language received a score of 0.77.[9] Four models are trained for experiments using three distinct feature sets taken from the dataset. The approaches to identify hate speech in videos involve converting the video into text format before providing it as input to machine learning models. The models are evaluated by computing the specified evaluation metrics..[10] explains how to add text data to supervised NLP problems, using the classification of deep internet hate speech as an example. produces a noticeable improvement in the detection of multiple hate speech classes, outperforming the baseline in the biggest online collection of hate speech by an absolute increase in Macro-F1 score of 5.7% and a 30% increase in hate speech class recall..[11] The suggested method uses sentimental and semantic features to classify tweets into hateful, offensive, and clean by automatically identifying hate speech patterns and the most prevalent unigrams..[12] Crowdsourcing was used to categorize a portion of those comments into three distinct groups: those having messages of hatred, primarily words that are offensive, as well as those missing both of them. A multi-class classifier was trained to differentiate between the three categories. Lexical techniques are useful for locating possibly offensive terms.

III.Proposed Work

Data Pre-Processing

Pre-processing is the term used to describe the modifications made to our data before we send it to the algorithm. Data Pre processing is a technique for turning incomplete data into a complete set of data. In other words, data is always gathered from different sources in an unprocessed form that precludes analysis. For the machine learning technique used to apply the model to produce improved results, the data must be properly organised. The Random Forest algorithm, for example, does not allow null numbers, so some machine learning models have particular information requirements. In order to use the random forest algorithm, null values from the original raw data collection must be managed. A single dataset should be able to run numerous Deep Learning and machine learning algorithms thanks to the data collection's format.

Data validation/cleaning process

The process of validating data that has been collected involves verifying and validating it before use. As a result, it implies that source data must be checked for accuracy and quality before being used, imported, or processed in any other way. Data cleaning's primary goal is to find and eliminate mistakes and anomalies in order to improve the intrinsic value of data for analytics and decision-making.

Data Visualization

The visualization of data offers a crucial collection of techniques for achieving a qualitative grasp. The following may come in beneficial if looking into and learning about a dataset to find trends, inaccurate data, anomalies, as well as a lot more. In comparison to evaluations of correlation and importance, visualizations of data are able to be applied to articulate and demonstrate significant connections in graphs and visuals which make the information more tactile and compelling to stakeholders.

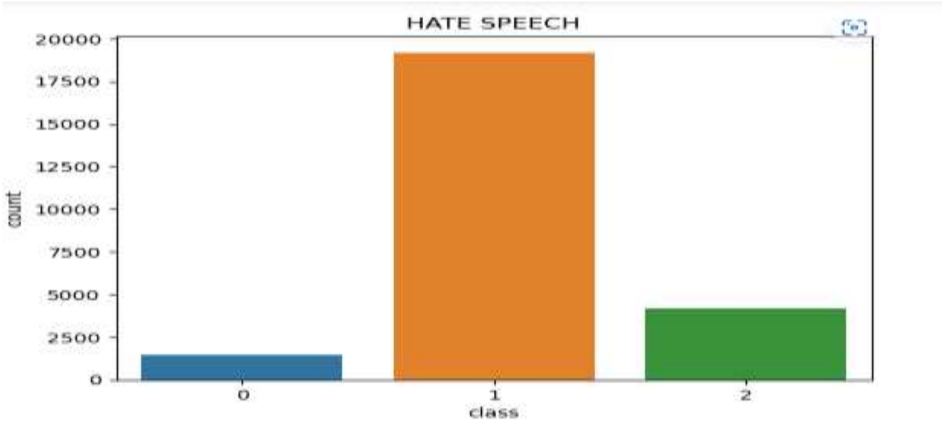

Fig 1. Data visualization graph

This image displays a data visualisation graph that displays the data as a bar graph. The X-axis in this case depicts the classes, which are 0, 1, and 2, and the Y-axis the count values.

Machine Learning Model Development

It will become clear that scikit-learn in Python can be used to build a test harness for this reason. Reliable comparison of the performance of numerous algorithms for machine learning is essential. This test harness can be used as a reference for your particular machine learning issues, along with additional and different ways of comparison. The efficiency characteristics of each type will differ. The potential accuracy of each model on unobserved data can be assessed by employing revising techniques as cross-validations. Out of the collection of models you've created using these approximations, it must be able to choose one or two of the finest ones. To see the data from different perspectives, it is a good idea to visualise new information using a variety of techniques. The model selection follows the same reasoning. You should evaluate the anticipated accuracy of your machine learning algorithms in a number of methods before choosing the one or two that will be used for finalisation. One method to do this is to display the average accuracy, variance, and other properties of the range of model accuracies using various visualisation techniques.

1.Random Forest Algorithm

Prominent machine learning algorithm called random forest remains an element of the guided learning methodology. This could be utilized to solve ML problems incorporating both regression and classification. This depends on the premise of ensemble learning, which is a technique for combining different classifiers to deal with complicated problems and improve model performance. As the acronym signifies, Random Forest acts as a classifier that averages several decision trees applied to various sections of the supplied dataset in order to boost the predictive ability of the dataset.

2.Naive Bayes Algorithm

The Naive Bayes algorithm is a straightforward method that uses probabilities for each trait that belongs to each class to make predictions. It is the supervised learning approach you would use to model a forecast modelling problem probabilistically. The naive bayes approach simplifies the computation of probabilities by presuming that the likely outcome of every feature being associated with an identified group result is distinct from that of the remaining features. Despite being a powerful assumption, this one results in a prompt and effective method. Naive Bayes is the name of a statistical categorization technique dependent upon Bayes Theorem. This constitutes one of several the most straightforward guided learning techniques. The naive Bayes classification algorithm is the most efficient, reliable, and quick method. Naive Bayes classifiers function swiftly and precisely on large datasets.

3.Gradient Boosting

One of the well-liked learning ensemble modeling methods called "boosting" is used to create strong classifiers from a variety of weak classifiers. It begins by creating a primary model using training data sets that are readily accessible, and then it finds any errors in the base model. A secondary model is constructed after the error has been located, and a third model is then added to the procedure. In this manner, adding additional models is continued until we have a complete collection of training data from which the model can accurately predict. In the annals of machine learning, GBM is also used as an ensemble technique to transform weak learners into strong ones. In this topic, "GBM in Machine Learning," we'll talk about boosting algorithms, gradient machine learning algorithms, the history of GBM, how it functions, different GBM terminologies, etc. But first, familiarize yourself with the boosting idea and different boosting algorithms used in machine learning.

DEPLOYMENT USING FLASK:

The most accurate algorithm is used to implement the model Gradient Boosting, one of three algorithms, is the strongest model. The Flask micro web frame work is used to deploy this model.

IV.Evaluation Metrics

Classification Report

In machine learning, a classification report is a success evaluation metric. It is used to display the trained classification model's accuracy, recall, F1 Score, and support. The classification reports for the three methods are as follows:

i. Random Forest Algorithm

```
from sklearn.metrics import classification_report
print('Classification report of Random forest Classifier\n\n',classification_report(y_test,predict))

Classification report of Random forest Classifier

                        precision    recall  f1-score   support

           Hate_Speech       0.71      0.62      0.67        16
No_Hate_and_Offensive       0.81      0.88      0.85        25
   Offensive_Language       0.84      0.84      0.84        19

              accuracy                           0.80        60
             macro avg       0.79      0.78      0.78        60
          weighted avg       0.80      0.80      0.80        60
```

Fig 2. Screenshot of classification report of Random forest algorithm

The aforementioned figure displays the precision, recall, f1-score, and support values for hate speech, no hate and offensive language, and offensive language that were obtained by using the Random Forest method to train the model.

ii. Naïve Bayes Algorithm

```
from sklearn.metrics import classification_report
print('Classification report of Multinomial Naive Bayes\n\n',classification_report(y_test,predict))

Classification report of Multinomial Naive Bayes

                        precision    recall  f1-score   support

           Hate_Speech       0.72      0.76      0.74        41
No_Hate_and_Offensive       0.94      0.76      0.84        41
   Offensive_Language       0.77      0.89      0.83        38

              accuracy                           0.80       120
             macro avg       0.81      0.80      0.80       120
          weighted avg       0.81      0.80      0.80       120
```

Fig 3. Screenshot of classification report of Naïve Bayes algorithm

The accuracy, recall, f1-score, and support value for Hate_Speech, No_Hate_and_Offensive, and Offensive Language as determined by model training with the Naive Bayes Algorithm are shown in the above figure.

iii. Gradient Boosting Algorithm

```
from sklearn.metrics import classification_report
print('Classification report of Gradient Boosting Classifier\n\n',classification_report(y_test,predict))

Classification report of Gradient Boosting Classifier

                        precision    recall  f1-score   support

           Hate_Speech       0.73      0.80      0.76        10
No_Hate_and_Offensive       0.89      1.00      0.94        17
   Offensive_Language       0.92      0.75      0.83        16

              accuracy                           0.86        43
             macro avg       0.85      0.85      0.84        43
          weighted avg       0.87      0.86      0.86        43
```

Fig 4. Screenshot of classification report of Gradient Boosting algorithm

The precision, recall, f1-score, and support value for hate speech, no hate and offensive language are displayed in the above figure. These values were acquired by using the Gradient Boosting Algorithm to train the model.

Confusion Matrix

The performance of the classification models for a certain set of test data is evaluated using a matrix called the confusion matrix. It can only be established if the real test data values are known.

i. Random Forest Algorithm

```
from sklearn.metrics import confusion_matrix
print('confusion matrix of Random forest Classifier\n\n',confusion_matrix(y_test,predict))
```

```
confusion matrix of Random forest Classifier

 [[2 0 0]
 [0 4 0]
 [2 0 2]]
```

Fig 5. Screenshot of confusion matrix of Random forest algorithm

For Hate Speech the TruePositive Value is 2.The FalseNegative for hate speech is [0+0]=0. The FalsePositive for hate speech is [0+2]=2.TrueNegative value for hate speech is [4+0+0+2]=6.

ii. Naïve Bayes Algorithm

```
from sklearn.metrics import confusion_matrix
print('confusion matrix of Multinomial Naive Bayes\n\n',confusion_matrix(y_test,predict))
```

```
confusion matrix of Multinomial Naive Bayes

 [[31  2  8]
 [ 8 31  2]
 [ 4  0 34]]
```

Fig 6. Screenshot of confusion matrix of Naïve Bayes algorithm

For Hate Speech the TruePositive Value is 31. The FalseNegative for hatespeech is [2+8] = 10. The FalsePositive for hate speech is [8+4]=12. TrueNegative value for hate speech is [31+2+0+34]=67.

iii. Gradient Boosting Algorithm

```
from sklearn.metrics import confusion_matrix
print('confusion matrix of Gradient Boosting Classifier\n\n',confusion_matrix(y_test,predict))
```

```
confusion matrix of Gradient Boosting Classifier

 [[ 8  1  1]
 [ 0 17  0]
 [ 3  1 12]]
```

Fig 7. Screenshot of confusion matrix of Gradient Boosting algorithm

For Hate Speech the TruePositive Value is 8. The FalseNegative for hate speech is [1+1]=2. The FalsePositive for hate speech is [0+3]=3.TrueNegative value for hate speech is [17+0+1+12]=30.

`

V.Accuracy Comparison

The Gradient Boosting algorithm is determined to provide the best accuracy of the three algorithms. As a result, the project's implementation used the Gradient boosting algorithm.

i. Random Forest Algorithm

```
from sklearn.metrics import accuracy_score
print('Accuracy of Random forest Classifier is ',accuracy_score(y_test,predict)*100)
```

```
Accuracy of Random forest Classifier is  80.0
```

Fig 8. Screenshot of accuracy of Random forest algorithm

The accuracy achieved by using the Random Forest algorithm during model training is shown in the above image. The degree of precision found is, that is 80.0% of random forest classifiers are accurate.

ii. Naïve Bayes Algorithm

```
from sklearn.metrics import accuracy_score
print('Accuracy of Multinomial Naive Bayes is ',accuracy_score(y_test,predict)*100)
```

```
Accuracy of Multinomial Naive Bayes is  80.0
```

Fig 9. Screenshot of accuracy of Naïve Bayes Algorithm

The accuracy achieved by using the Naive Bayes algorithm during model training is shown in the above image. The accuracy found is 80.0%, which is the accuracy of the multinomial naive bayes model.

iii. Gradient Boosting Algorithm

```
from sklearn.metrics import accuracy_score
print('Accuracy of Gradient Boosting Classifier is ',accuracy_score(y_test,predict)*100)
```

```
Accuracy of Gradient Boosting Classifier is  86.04651162790698
```

Fig 10. Screenshot of accuracy of Gradient Boosting

The accuracy achieved by using the Gradient Boosting algorithm to train the model is shown in the above image. The accuracy attained is 86.04%, which is the accuracy of the gradient-boosting classifier.

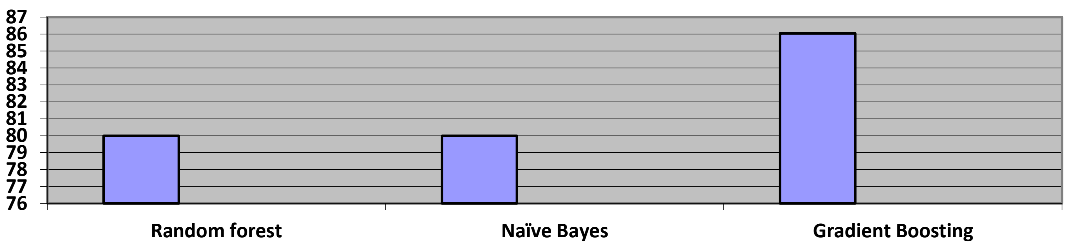

Fig 11. Accuracy Comparison

Three suggested work algorithms are compared in the graph up top. According to the information above, "Gradient Boosting Algorithm" produces the greatest accuracy when compared to the other two. The Gradient Boosting Algorithm, which produces the greatest accuracy, is now used in the project deployment.

Comparison of accuracies in proposed and related work:

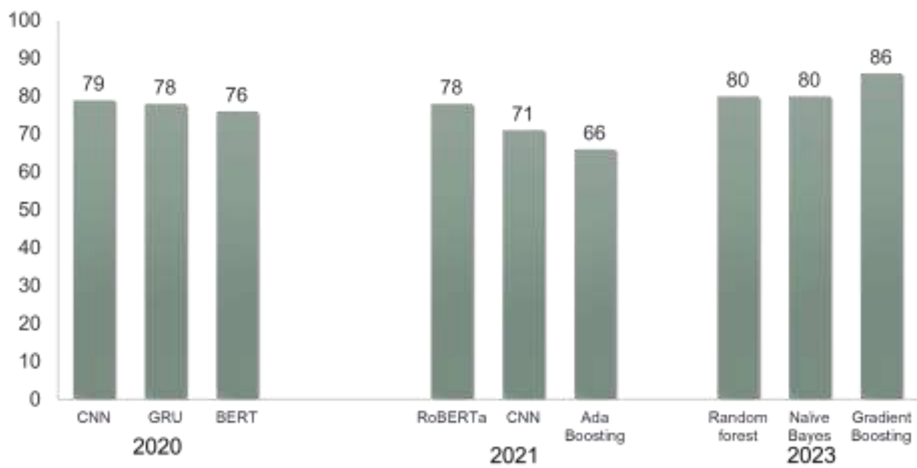

Fig 12. Comparison of accuracies in proposed and related work

According to the performance analysis presented above, the Gradient Boosting Algorithm produces the greatest accuracy. The accuracy of algorithms like CNN, GRU, and BERT in study papers from 2020 was 79%, 78%, and 76%, respectively. The accuracy of algorithms like RoBERTa, CNN, and Ada Boosting in study papers from 2021 was 78%, 71%, and 66%, respectively. This leads to the conclusion that the proposed work has produced better accuracy results than its related work.

VI. Results and Discussions

Fig 13. Represents the user interface where users enter comments for hate speech, offensive language, no hatred, and offensive classification.

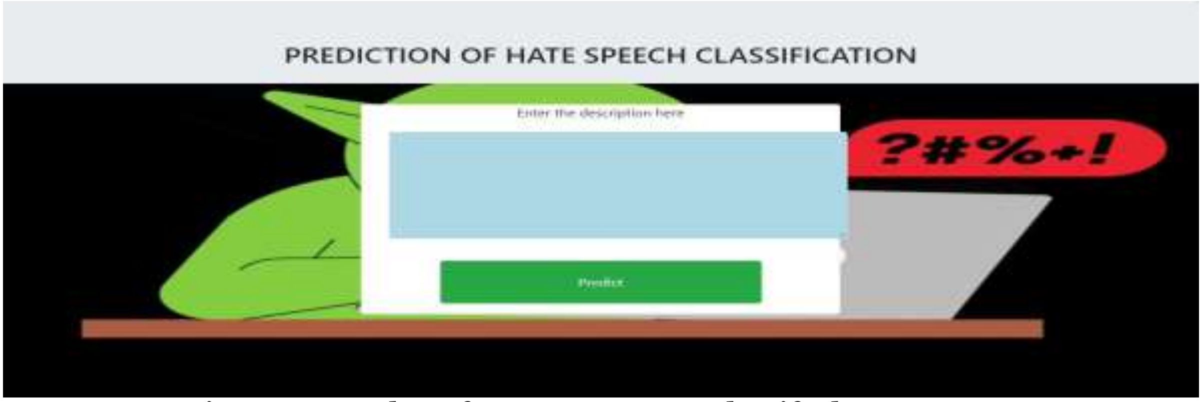

Fig 13. Screenshot of output screen to classify the comments

This screenshot displays the home page, where text is entered and remarks are organised.

Fig 14. Screenshot of entering the text for classification

This figure demonstrates how to type comments into the textbox to input text.The "Predict" button should be clicked after entering the text in the textbox to reveal the sort of comment.

Fig 15. Screenshot of the prediction of Hate Speech

When the "Predict" button is selected after the text has been entered, this figure illustrates how the comments will be categorised and presented in the webpage. The sort of comment displayed in this figure is hate speech.

VII. Conclusion and Future Enhancements

As the world moves more and more toward digital technology, everyone now has access to the internet and can publish whatever they want, making the control of hate speech very difficult. Therefore, there is a higher likelihood that individuals will be misled. Machine learning is typically designed to handle these types of complex tasks because it requires more time to manually analyze these types of data. Machine learning can be used to determine whether or not a piece of speech qualifies as hate speech by using historical data, helping them understand trends, and increasing the model's accuracy by tweaking its parameters. After that, the model is employed as a categorization model. The optimal model can be used for classification by comparing various algorithms. The Gradient Boosting Algorithm is determined to provide the best accuracy of the three algorithms. As a result, the project's implementation used the Gradient boosting algorithm. In the future, if someone attempts to use any offensive language, a warning notice will be displayed, and if they go over the allotted number of warnings, their account will be blocked from that particular social network.

References

1. Soumitra Ghosh , Asif Ekbal , Pushpak Bhattacharyya, Tista Saha, Alka Kumar, and Shikha Srivastava,"SEHC: A Benchmark Setup to Identify Online Hate Speech in English" in IEEE Transactions on Computational Social Systems, 2022
2. Zewdie Mossie, Jenq-Haur Wang,"Vulnerable community identification using hate speech detection on social media" in Information Processing & Management 57 (3), 102087, 2020
3. Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, Animesh Mukherjee," HateXplain "A Benchmark Dataset for Explainable Hate Speech Detection" in Proceedings of the AAAI Conference on Artificial Intelligence 35 (17), 14867-14875, 2021
4. Kovács Pedro Alonso Rajkumar Saini," Challenges of Hate Speech Detection in Social Media" in SN Computer Science 2, 1-15, 2021
5. Amrutha, B R and Bindu, K R, "Detecting Hate Speech in Tweets Using Different Deep Neural Network Architectures in 2019 International Conference on Intelligent Computing and Control Systems (ICCS), 923-926, 2019
6. Sindhu Abro1 , Sarang Shaikh2 , Zafar Ali ," Automatic Hate Speech Detection using Machine Learning: A Comparative Study"in International Journal of Advanced Computer Science and Applications 11 (8), 2020
7. Raghad Alshalan and Hend Al-Khalifa," A Deep Learning Approach for Automatic Hate Speech Detection in the Saudi Twittersphere"in Applied Sciences 10 (23), 8614, 2020
8. Varsha Pathak , Manish Joshi , Prasad A. Joshi , Monica Mundada , Tanmay Joshi," Using Machine Learning for Detection of Hate Speech and Offensive Code-Mixed Social Media text " in arXiv preprint arXiv:2102.09866, 2021
9. Ching Seh Wu, Unnathi Bhandary," Detection of Hate Speech in Videos Using Machine Learning" in 2020 International Conference on Computational Science and Computational Intelligence (CSCI), 585-590, 2020
10. Abd Algani, Y. M., Caro, O. J. M., Bravo, L. M. R., Kaur, C., Al Ansari, M. S., & Bala, B. K. (2023). Leaf disease identification and classification using optimized deep learning. *Measurement: Sensors*, *25*, 100643.
11. J. K. S. Al-Safi, A. Bansal, M. Aarif, M. S. Z. Almahairah, G. Manoharan and F. J. Alotoum, "Assessment Based On IoT For Efficient Information Surveillance Regarding Harmful Strikes Upon Financial Collection," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-5, doi: 10.1109/ICCCI56745.2023.10128500.
12. Kaur, C., Kumar, M. S., Anjum, A., Binda, M. B., Mallu, M. R., & Al Ansari, M. S. (2023). Chronic kidney disease prediction using machine learning. *Journal of Advances in Information Technology*, *14*(2), 384-391.
13. Eadline, D. (2015). *Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem*. Addison-Wesley Professional.
14. M. A. Tripathi, R. Tripathi, F. Effendy, G. Manoharan, M. John Paul and M. Aarif, "An In-Depth Analysis of the Role That ML and Big Data Play in Driving Digital Marketing's Paradigm Shift," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-6, doi: 10.1109/ICCCI56745.2023.10128357.
15. Khan, S. I., Kaur, C., Al Ansari, M. S., Muda, I., Borda, R. F. C., & Bala, B. K. (2023). Implementation of cloud based IoT technology in manufacturing industry for smart control of manufacturing process. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 1-13.
16. M. Lourens, A. Tamizhselvi, B. Goswami, J. Alanya-Beltran, M. Aarif and D. Gangodkar, "Database Management Difficulties in the Internet of Things," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, 2022, pp. 322-326, doi: 10.1109/IC3I56241.2022.10072614.

17. Abd Algani, Y. M., Caro, O. J. M., Bravo, L. M. R., Kaur, C., Al Ansari, M. S., & Bala, B. K. (2023). Leaf disease identification and classification using optimized deep learning. *Measurement: Sensors*, *25*, 100643.
18. Ratna, K. S., Daniel, C., Ram, A., Yadav, B. S. K., & Hemalatha, G. (2021). Analytical investigation of MR damper for vibration control: a review. *Journal of Applied Engineering Sciences*, *11*(1), 49-52.
19. Naidu, K. B., Prasad, B. R., Hassen, S. M., Kaur, C., Al Ansari, M. S., Vinod, R., ... & Bala, B. K. (2022). Analysis of Hadoop log file in an environment for dynamic detection of threats using machine learning. *Measurement: Sensors*, *24*, 100545.
20. Abd Algani, Y. M., Ritonga, M., Kiran Bala, B., Al Ansari, M. S., Badr, M., & Taloba, A. I. (2022). Machine learning in health condition check-up: An approach using Breiman's random forest algorithm. *Measurement: Sensors*, *23*, 100406. https://doi.org/10.1016/j.measen.2022.100406
21. Mourad, H. M., Kaur, D., & Aarif, M. (2020). Challenges Faced by Big Data and Its Orientation in the Field of Business Marketing. *International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)*, *10*(3), 8091-8102.
22. Zainal, A. G. (2022). Recognition of Copy Move Forgeries in Digital Images using Hybrid Optimization and Convolutional Neural Network Algorithm. *(IJACSA) International Journal of Advanced Computer Science and Applications*, *13*(12).
23. Aarif, Mohd, and Ali Alalmai. "Importance of Effective Business Communication for promoting and developing Hospitality Industry in Saudi Arabia." *A case study of Gizan (Jazan)* (2019).
24. Georgios Rizos, Konstantin Hemker, Bjorn Schuller," Augment to Prevent: Short-Text Data Augmentation in Deep Learning for Hate-Speech Classification" in the Proceedings of the 28th ACM international conference on information and knowledge management, 991-1000, 2019
25. Hajime Watanabe, Mondher Bouazizi, Tomoaki Ohtsuki ," Hate Speech on Twitter: A Pragmatic Approach to Collect Hateful and Offensive Expressions and Perform Hate Speech Detection"in IEEE access 6, 13825-13835, 2018
26. Thomas Davidson,Dana Warmsle,Michael Macy,Ingmar Weber," Automated Hate Speech Detection and the Problem of Offensive Language" in Proceedings of the international AAAI conference on web and social media 11 (1), 512-515, 2017