

CLASSIFICATION OF ONLINE TOXIC COMMENTS USING NLP ALGORITHMS

K.VENKATESH¹, MARY PURILLA²¹Assistant Professor MCA, DEPT, Dantuluri Narayana Raju College , Bhimavaram, AndhrapradeshEmail id:- kornalavenkatesh@gmail.com²PG Student of MCA, Dantuluri Narayana Raju College , Bhimavaram, AndhrapradeshEmail id :- marypurilla101@gmail.com**ABSTRACT**

A large proportion of online comments present on public domains are usually constructive, however a significant proportion are toxic in nature. Dataset is obtained online which are processed to remove noise from the dataset. The comments contain lot of errors which increases the number of features manifold, making the machine learning model to train the dataset by processing the dataset, in the form of transformation of raw comments before feeding it to the Classification models using a machine learning technique known as the term frequency-inverse document frequency (TF-IDF) technique. The logistic regression technique is used to train the processed dataset, which will differentiate toxic comments from non-toxic comments. The multi-headed model comprises toxicity (severe toxic, obscene, threat, insult, and identity-hate) or Non-Toxicity Evaluation, using confusion metrics for their prediction.

1 INTRODUCTION

The exponential development of computer science and technology provides us with one of the greatest innovations of the "Internet" of the 21st century, where one person can communicate to another worldwide with the help of a mere smart phone and internet.

In the initial days of the internet, people used to communicate with each other through Email only and it was filled with spam emails. In those days, it was a big task to classify the emails as positive or negative i.e. spam or not - spam.

As time flows, communication, and flow of data over the internet got changed drastically, especially after the appearance of social media sites. With the advancement of social media, it becomes highly important to classify the content into positive and negative terms, to prevent any form of harm to society and to control antisocial behavior of people.

In recent times there have many instances where authorities arrest people due to their harmful and toxic social media contents.

For example, one 28-year-old man was arrested in Bengal for posting an abusive comment against Mamata Banerjee on Facebook and one man from Indonesia was arrested for insulting the police of Indonesia on Facebook. Thus, there is an alarming situation and it is the need of the hour to detect such content before they got published because these negative contents are creating the internet an unsafe place and affecting people adversely.

Suppose there is a comment on social media "Nonsense? Kiss off, geek. What I said is true", it can be easily identified that the words like Nonsense and Kiss off are negative and thus this comment is toxic. But to mine the toxicity technically this comment needs to go through a particular procedure and then classification technique will be applied on it to verify the precision of the obtained result.

Different machine learning algorithms will be used in the classification of toxic comments on the Data set of Kaggle.com. This paper includes six machine learning techniques i.e. logistic regression, random forest, SVM classifier, naive bayes, decision tree, and KNN classification to solve the problem of text classification. So, we will apply all the six machine learning algorithms on the given data set and calculate and compare their accuracy, log loss, and hamming loss.

2. LITERATURE SURVEY AND RELATED WORK**2.1 A Web of Hate: Tackling Hateful Speech in Online Social Spaces,**

H. M. Saleem, K. P . Dillon, S. Benesch, and D. Rut hs,

Online social platforms are beset with hateful speech - content that expresses hatred for a person or group of people. Such

content can frighten, intimidate, or silence platform users, and some of it can inspire other users to commit violence. Despite widespread recognition of the problems posed by such content, reliable solutions even for detecting hateful speech are lacking. In the present work, we establish why keyword-based methods are insufficient for detection. We then propose an approach to detecting hateful speech that uses content produced by self-identifying hateful communities as training data. Our approach bypasses the expensive annotation process often required to train keyword systems and performs well across several established platforms, making substantial improvements over current state-of-the-art approaches.

2.2 Corpus for research on deliberation and debate

M. A. Walker, P. Anand, J. E. F. Tree, R. Abbott, and J. King

Deliberative, argumentative discourse is an important component of opinion formation, belief revision, and knowledge discovery; it is a cornerstone of modern civil society. Argumentation is productively studied in branches ranging from theoretical artificial intelligence to political rhetoric, but empirical analysis has suffered from a lack of freely available, unscripted argumentative dialogs. This paper presents the Internet Argument Corpus (IAC), a set of 390,704 posts in 11,800 discussions extracted from the online debate site 4forums.com. A 2866 thread/130,206 post extract of the corpus has been manually sided for topic of discussion, and subsets of this topic-labeled extract have been annotated for several dialogic and argumentative markers: degrees of agreement with a previous post, cordiality, audience direction, combativeness, assertiveness, emotionality of argumentation, and sarcasm. As an application of this resource, the paper closes with a discussion of the relationship between discourse marker pragmatics, agreement, emotionality, and sarcasm in the IAC corpus.

2.3 Antisocial behavior in online discussion communities

J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec

User contributions in the form of posts, comments, and votes are essential to the success of online communities. However, allowing user participation also invites undesirable behavior such as trolling. In this paper, we characterize antisocial behavior in three large online discussion communities by analyzing users who were banned from these communities. We find that such users tend to concentrate their efforts in a small number of threads, are more likely to post irrelevantly, and are more successful at garnering responses from other users. Studying the evolution of these users from the moment they join a community up to when they get banned, we find that not only do they write worse than other users over time, but they also become increasingly less tolerated by the community. Further, we discover that antisocial behavior is exacerbated when community feedback is overly harsh. Our analysis also reveals distinct groups of users with different levels of antisocial behavior that can change over time. We use these insights to identify antisocial users early on, a task of high practical importance to community maintainers.

2.4 Abusive language detection in online user content

C. Nobata, J. Tet reault, A. Thomas, Y. Mehdad, and Y. Chang,

Detection of abusive language in user generated online content has become an issue of increasing importance in recent years. Most current commercial methods make use of blacklists and regular expressions, however these measures fall short when contending with more subtle, less ham-fisted examples of hate speech. In this work, we develop a machine learning based method to detect hate speech on online user comments from two domains which outperforms a state-of-the-art deep learning

approach. We also develop a corpus of user comments annotated for abusive language, the first of its kind. Finally, we use our detection tool to analyze abusive language over time and in different settings to further enhance our knowledge of this behavior.

2.5 Text Classification Using Machine Learning Techniques,

E. K. Ikonomakis, S. Kot siantis, and V. T ampakas

Text classification is a smart classification of text into categories. And, using machine learning to automate these tasks, just makes the whole process super-fast and efficient. Artificial Intelligence and Machine learning are arguably the most beneficial technologies to have gained momentum in recent times. They are finding applications everywhere. As Jeff Bezos said in his annual shareholder's letter.

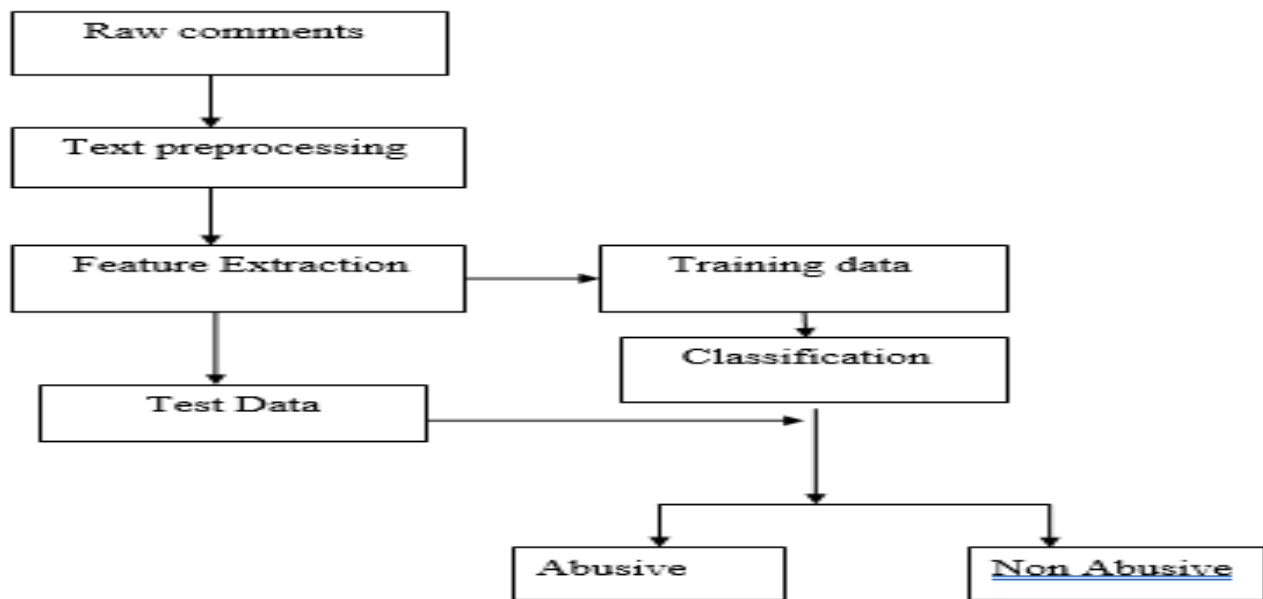


Fig 1: System Architecture

3 EXISTING SYSTEM

In The Existing system used Naive Bayes. In Naive Bayes, texts are classified based on posterior probabilities generated based on the presence of different classes of words in texts. This assumption makes the computations resources needed for a naive bayes classifier far more efficient than non-naive bayes approaches which are exponential in complexity. Moreover, it is found that Naive Bayes is the less accurate model for text classification.

3.1.1 Disadvantages of Existing system

- ⊗ The main limitation of Naive Bayes is the assumption of independent predictor features. Naive Bayes implicitly assumes that all the attributes are mutually independent.
- ⊗ In real life, it's almost impossible that we get a set of predictors that are completely independent or one another.
- ⊗ Less quality text classification by using naive bayes.
- ⊗ We haven't implemented tf-idf concept for classification

Algorithm: Naive bayes.

4 PROPOSED WORK AND ALGORITHM

The proposed method is based on the Random forest and is proposed to perform text classification. In the traditional random forest algorithm, the number and quality of feature selection are prominent. But for books and other large capacity text classification, the more the number and quality of text features (classification decision tree attribute), the better the classification effect will be. Therefore, this paper proposes a tr-k method which combines TF-IDF, text rank and K-means to improve the effect of text classification. The full name of the TF-IDF method is term frequency inverse document frequency.

3.2.1 Advantages of Proposed system

Random forests overcome several problems with decision trees, including:

- ⊗ Reduction in over fitting: by averaging several trees, there is a significantly lower risk of over fitting.
- ⊗ Less variance: By using multiple trees, you reduce the chance of stumbling across a classifier that doesn't perform well because of the relationship between the train and test data.
- ⊗ tr-k method which combines TF-IDF, textrank and K-means to improve the effect of text classification.
- ⊗ RFA has achieved good results in biochip, information extraction and other fields.

Algorithm: Random

5 METHODOLOGIES

MODULES

- **Data Preprocessing:** This module involves cleaning and preprocessing the comment data, including removing irrelevant characters, converting text to lowercase, and tokenization.
- **Embedding Layer:** The pre-trained Glove embeddings are used to represent words in a continuous vector space, capturing semantic relationships between words.
- **Convolutional Neural Network (CNN):** The CNN model is employed to extract relevant features from the comment data by applying convolutional filters.
- **Classification Layer:** The extracted features are then fed into a classification layer that predicts the toxicity or negativity of the comments based on the learned patterns.
- **Optimization:** Grid search is used to optimize hyper parameters such as batch size, number of epochs, learning rate, dropout rate, and filter size, to enhance the model's performance.
- **Performance Evaluation:** Confusion matrices are generated separately for each of the six labels to evaluate the performance of the system in classifying comments.

By adopting the proposed system, online platforms can effectively identify and moderate toxic or negative comments, creating a safer and more inclusive online community for users.

6 RESULTS AND DISCUSSION



Fig 1: In above screen selecting and uploading 'train.csv' file and then click on 'Open' button to load dataset and to get below screen



Fig 2: In above screen dataset loaded and now click on "Preprocess Dataset" button to read dataset and then clean it.

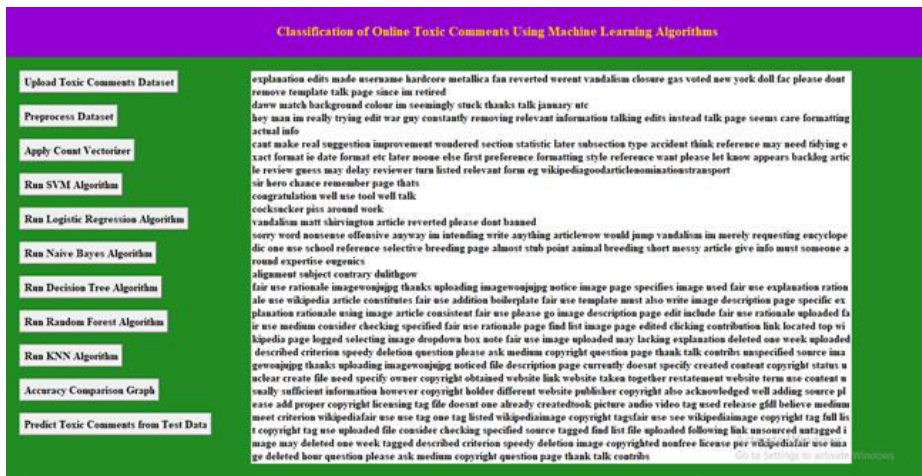


Fig 3: In above screen in text area we can see all comments are read and then clean and displaying them and now click on 'Apply Count Vectorizer' button to count each word and build a vector

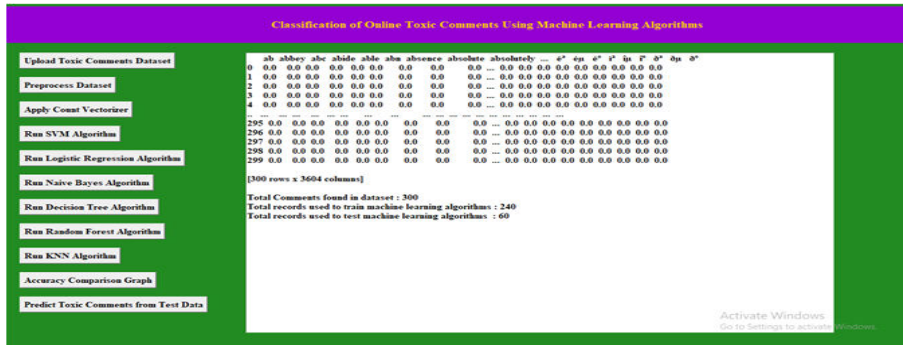


Fig 4: In above screen we can see vector is generated and in first row we can see words names and in remaining rows we can see their count and if word not appear in comments then 0 will be put. Now in above screen we are displaying only few records. Now train vector is ready and now click on 'Run SVM Algorithm' button to train SVM with above dataset and in above screen we can see application using 240 records for training and 60 records for testing.

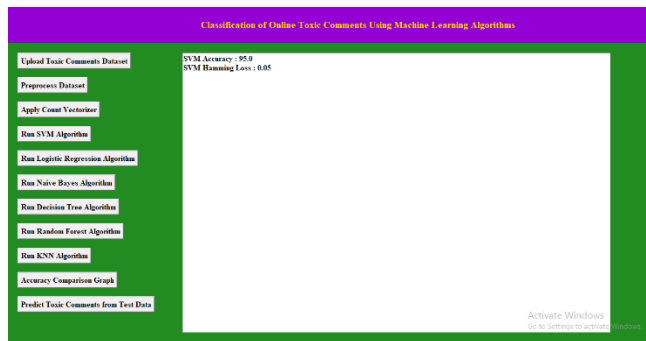


Fig 5: In above screen SVM ML model build with accuracy as 95% and loss as 0.05% and similarly click all algorithms button to train ML model for each algorithm

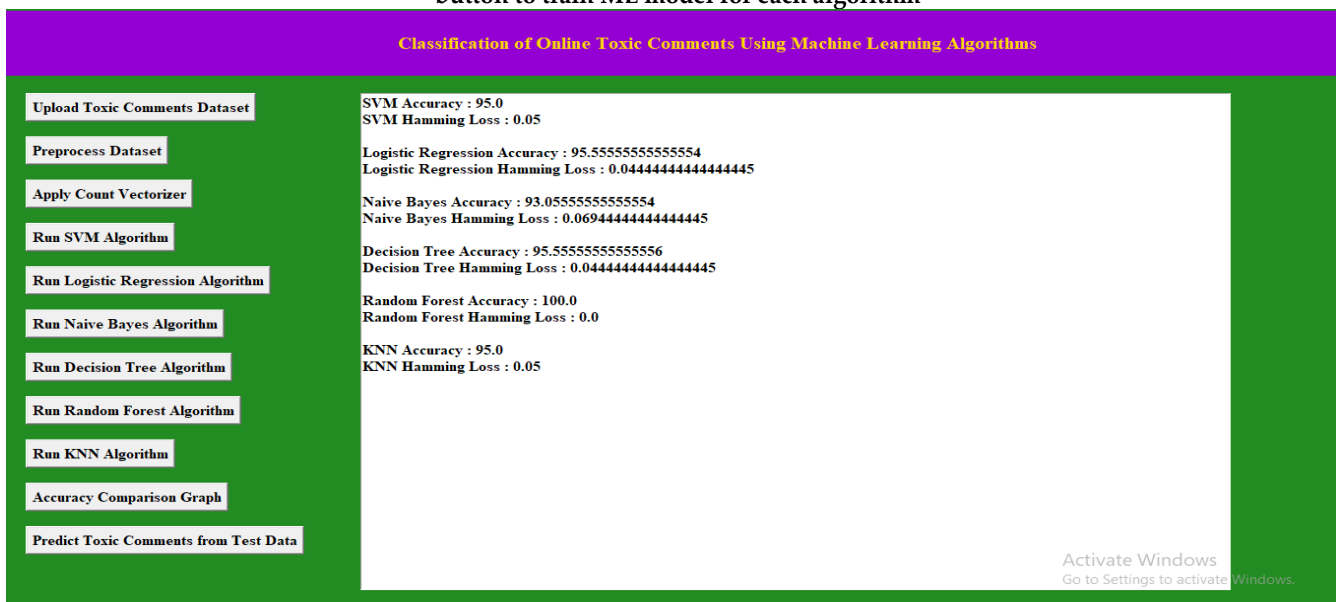


Fig 6: In above screen we can see accuracy and loss value for all algorithms and in above screen random forest gave 100% accuracy with 0% as loss and now click on 'Accuracy Comparison Graph' button to get below graph.

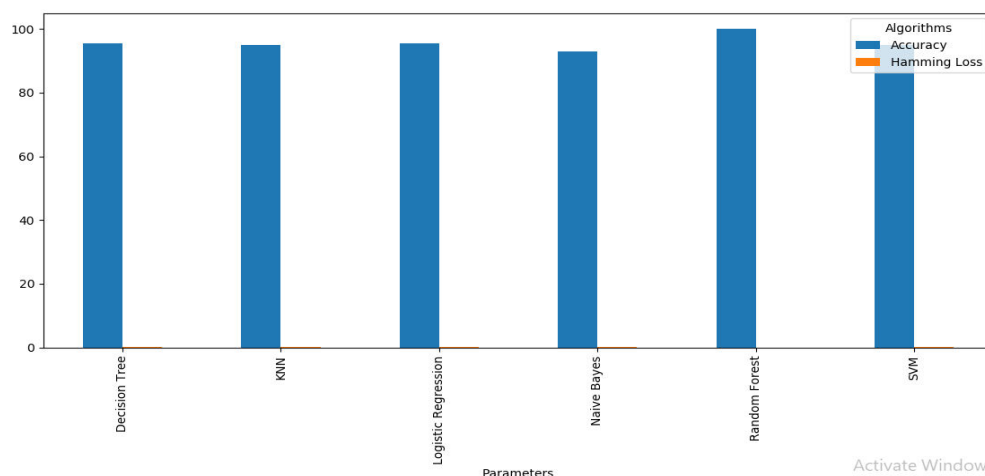


Fig 7: In above graph x-axis represents algorithm name and y-axis represents accuracy and loss value and in above graph all algorithms gave accuracy closer to 100% with minor loss value so loss is not plotting in graph. Now click on 'Predict Toxic Comments from Test Data' button to upload test data and then ML will predict comments are toxic or non-toxic.

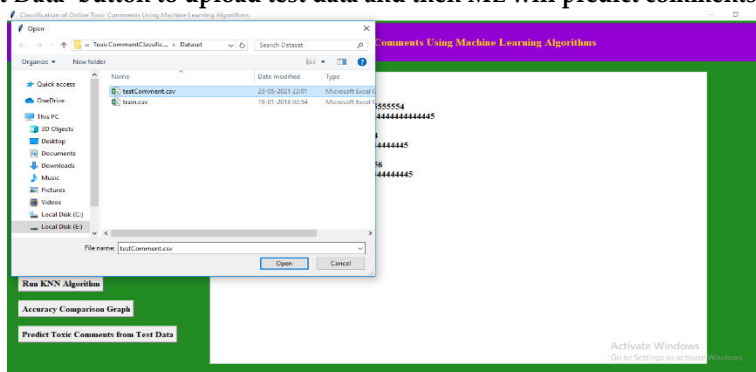


Fig 8: In above screen selecting and uploading 'testComment.csv' and then click on 'Open' button to get below prediction output.



Fig 9: In above screen first we are displaying comments and then in square bracket we are displaying predicted result as '[Contains TOXIC Comments]' or '[NOT CONTAINS TOXIC Comments]'.

6. CONCLUSION AND FUTURE SCOPE

Conclusion

We have discussed six Machine learning techniques i.e. logistic regression, Naive Bayes, decision tree, random forest, KNN classification, and SVM classifier, and compared their hamming loss, accuracy, and log loss in this paper. Now after proper

analysis, we can say that in terms of hamming loss, logistic regression performs best because in that case, our hamming loss is least, while in terms of accuracy, logistic regression performs best because accuracy is best in that model in comparison to other ones and terms of log loss, random forest works best due to least possible log loss in that model. So, our final model selection will be based on the combination of hamming loss and accuracy. Since we got the maximum accuracy i.e. 89.46 % and least possible hamming loss i.e. 2.43 % in case of the logistic regression model. We will select the logistic regression model as our final machine learning technique since it works best for our data.

Future Scope

- ⊙ Future scope for this analysis would be Integrating such classification algorithms into social media platforms to Automatically classify toxic comments .
- ⊙ To improve its accuracy and expand its application to different platforms and languages.
- ⊙ Helps to create a safer and more positive online environment by reducing the impact of Toxic behavior.

7 REFERENCES

- 1.H. M. Saleem, K. P . Dillon, S. Benesch, and D. Rut hs, "A Web of Hat e: T ackling Hat eful Speech in Online Social Spaces," 2017,[Online]Available: <http://arxiv.org/abs/1709.10159>.
2. M. Duggan, " Online harassment 2017," Pew Res., pp. 1 –85, 2017, doi: 202.419.4372.
3. M. A. Walker, P . Anand, J. E. F. T ree, R. Abbot t , and J. King, " A corpus for research on deliberat ion and debat e," P roc. 8t h Int . Conf. Lang. Resour. Eval. Lr. 2012, pp. 812–817, 2012.
4. J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec, " Ant isocial behavior in online discussion communities," P roc. 9t h Int . Conf. Web Soc. Media, ICWSM 2015, pp. 61–70, 2015.
5. B. Mat hew et al., "Thou shalt not hate: Countering online hate speech," P roc. 13th Int. Conf. Web Soc. Media, ICWSM 2019, no. August , pp. 369–380, 2019.
6. C. Nobata, J. Tet reault , A. Thomas, Y. Mehdad, and Y. Chang, " Abusive language det ect ion in online user cont ent ," 25t h Int . World Wide Web Conf. WWW 2016, pp. 145–153, 2016, doi: 10.1145/2872427.2883062.
7. E. K. Ikonomakis, S. Kot siantis, and V. T ampakas, "Text Classificat ion Using Machine Learning T echniques," no. August , 2005.
8. M. R. Murt y, J. V. . Murt hy, and P . Reddy P .V.G.D, " Text Document Classificat ion basedon Least Square Support Vector Machines with Singular Value Decomposit ion," Int. J. Comput . Appl., vol. 27, no. 7, pp. 21–26, 2011, doi: 10.5120/3312-4540.