MACHINE LEARNING APPROACH FOR CLASSIFICATION OF ONLINE TOXIC COMMENTS

[1] J.Sravanthi, Assistant Professor, Department of CSE, Chalapathi Institute of Technology, Guntur.
[2] Ikkurthi Lakshmanachari, B.Tech, Department of CSE, Chalapathi Institute of Technology, Guntur.
[3] Kambhampati Dilip, B.Tech, Department of CSE, Chalapathi Institute of Technology, Guntur.
[4] Komera Krupavathi, B.Tech, Department of CSE, Chalapathi Institute of Technology, Guntur.
[5] Kanneboyina Harika, B.Tech, Department of CSE, Chalapathi Institute of Technology, Guntur.

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[1]. 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

1) A Web of Hat e: T ackling Hat eful Speech in Online Social Spaces, H. M. Saleem, K. P . Dillon, S. Benesch, and D. Rut hs,

Abstract: 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) corpus for research on deliberat ion and debate M. A. Walker, P . Anand, J. E. F. T ree, R. Abbot t , and J. King

Abstract: 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

3) Antisocial behavior in online discussion communities J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec
Abstract: 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.

4) Abusive language det ect ion in online user content,C. Nobata, J. Tet reault , A. Thomas, Y. Mehdad, and Y. Chang,
Abstract: 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.

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 naïve bayes classifier far more efficient than non-naïve bayes approaches which are exponential in complexity. Moreover, it is found that Naive Bayes is the Less accurate model for text classification.

DISADVANTAGES

- 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

4. PROPOSED SYSTEM

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

ADVANTAGES

- 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.

SYSTEM ARCHITECTURE

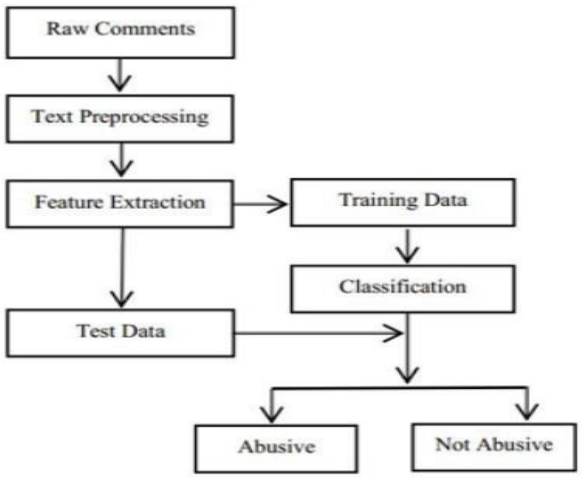


Fig : Flow chart for detecting toxic comments

5. ALGORITHMS

5.1 DECISION TREE CLASSIFIERS

Decision tree classifiers are used successfully in many diverse areas. Their most important feature is the capability of capturing descriptive decision making knowledge from the supplied data. Decision tree can be generated from training sets. The procedure for such generation based on the set of objects (S), each belonging to one of the classes C1, C2, …, Ck is as follows:

Step 1. If all the objects in S belong to the same class, for example Ci, the decision tree for S consists of a leaf labeled with this class

Step 2. Otherwise, let T be some test with possible outcomes O1, O2,…, On. Each object in S has one outcome for T so the test partitions S into subsets S1, S2,… Sn where each object in Si has outcome Oi for T. T becomes the root of the decision tree and for each outcome Oi we build a subsidiary decision tree by invoking the same procedure recursively on the set Si.

5.2 LOGISTIC REGRESSION CLASSIFIERS

Logistic regression analysis studies the association between a categorical dependent variable and a set of independent (explanatory) variables. The name logistic regression is used when the dependent variable has only two values, such as 0 and 1 or Yes and No. The name multinomial logistic regression is usually reserved for the case when the dependent variable has three or more unique values, such as Married, Single, Divorced, or Widowed. Although the type of data used for the dependent variable is different from that of multiple regression, the practical use of the procedure is similar. Logistic regression competes with discriminate analysis as a method for analyzing categorical-response variables. Many statisticians feel that logistic regression is more versatile and better suited for modeling most situations than is discriminate analysis. This is because logistic regression does not assume that the independent variables are normally distributed, as discriminate analysis does. This program computes binary logistic regression and multinomial logistic regression on both numeric and categorical independent variables. It reports on the regression equation as well as the goodness of fit, odds ratios, confidence limits, likelihood, and deviance. It performs a comprehensive residual analysis including diagnostic residual reports and plots. It can perform an independent variable subset selection search, looking for the best regression model with the fewest independent variables. It provides confidence intervals on predicted values and provides ROC curves to help determine the best cutoff point for classification. It allows you to validate your results by automatically classifying rows that are not used during the analysis.

5.3 SVM

In classification tasks a discriminate machine learning technique aims at finding, based on an independent and identically distributed (iid) training dataset, a discriminate function that can correctly predict labels for newly acquired instances. Unlike generative machine learning approaches, which require computations of conditional probability distributions, a discriminate classification function takes a data point x and assigns it to one of the different classes that are a part of the classification task. Less powerful than generative approaches, which are mostly used when prediction involves outlier detection, discriminate approaches require fewer computational resources and less training data, especially for a multidimensional feature space and when only posterior probabilities are needed. From a geometric perspective, learning a classifier is equivalent to finding the equation for a multidimensional surface that best separates the different classes in the feature space. SVM is a discriminate technique, and, because it solves the convex optimization problem analytically, it always returns the same optimal hyper plane parameter—in contrast to genetic algorithms (GAs) or perceptions, both of which are widely used for classification in machine learning. For perceptions, solutions are highly dependent on the initialization and termination criteria. For a specific kernel that transforms the data from the input space to the feature space, training returns uniquely defined SVM model parameters for a given training set, whereas the perception and GA classifier models are different each time training is initialized. The aim of GAs and perceptions is only to minimize error during training, which will translate into several hyper planes' meeting this requirement.

5.4 RANDOM FOREST

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned. Random decision forests correct for decision trees' habit of overfitting to their training set. Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees. However, data characteristics can affect their performance. The first algorithm for random decision forests was created in 1995 by Tin Kam Ho[1] using the

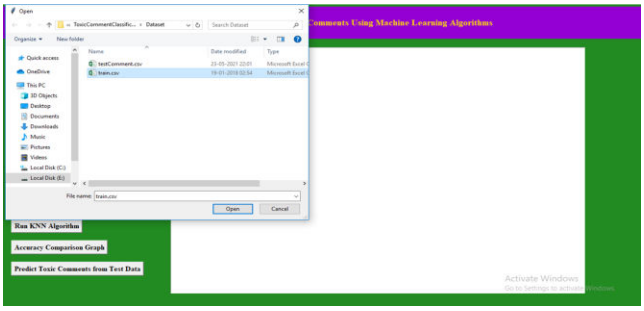
random subspace method, which, in Ho's formulation, is a way to implement the "stochastic discrimination" approach to classification proposed by Eugene Kleinberg. An extension of the algorithm was developed by Leo Breiman and Adele Cutler, who registered "Random Forests" as a trademark in 2006 (as of 2019, owned by Minitab, Inc.).The extension combines Breiman's "bagging" idea and random selection of features, introduced first by Ho[1] and later independently by Amit and Geman[13] in order to construct a collection of decision trees with controlled variance. Random forests are frequently used as "blackbox" models in businesses, as they generate reasonable predictions across a wide range of data while requiring little configuration.

6. RESULTS

6.1 Output Screens

To run this project double click on 'run.bat' file to get below screen

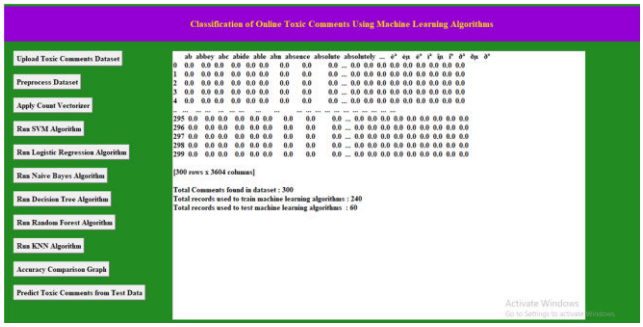
In above screen click on 'Upload Toxic Comments Dataset' button to upload dataset



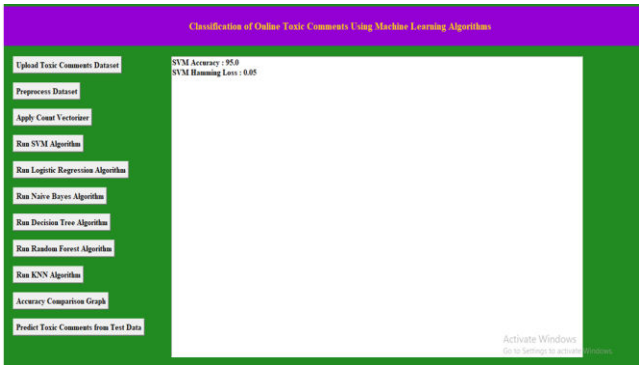
In above screen selecting and uploading 'train.csv' file and then click on 'Open' button to load dataset and to get below screen.

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

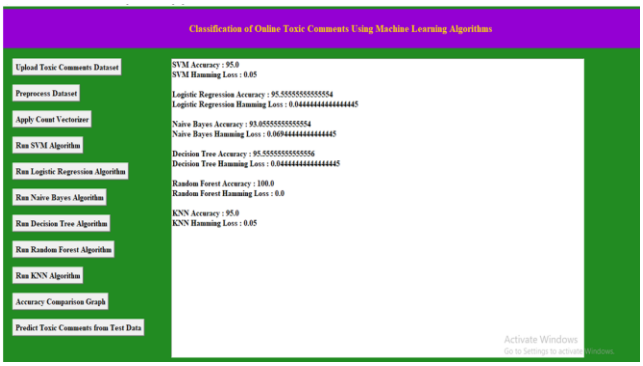
In above screen in text area we can see all comments are read and then clean and displaying them and now click on 'Apply Count Vectorizer" button to count each word and build a vector



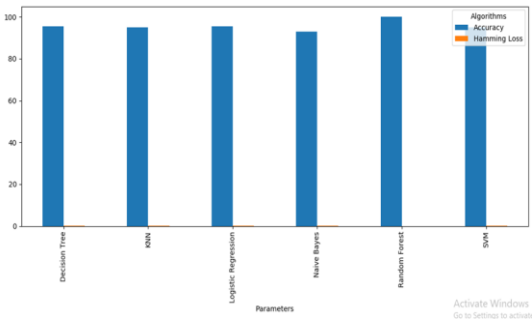
In above screen we can see vector is generated and in first row we can see words names and in remaining rows we can see their count and if word not appears in comments then 0 will be put. Now in above screen we are displaying only few records. Now train vector is ready and now click on 'Run SVM Algorithm' 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



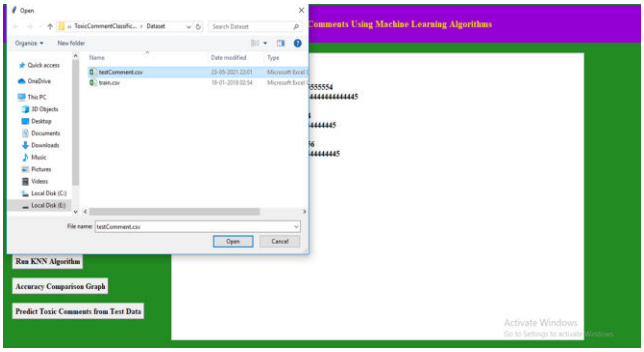
In above screen SVM ML model build with accuracy as 95% and loss as 0.05% and similarly click all algorithms button to train ML model for each algorithm



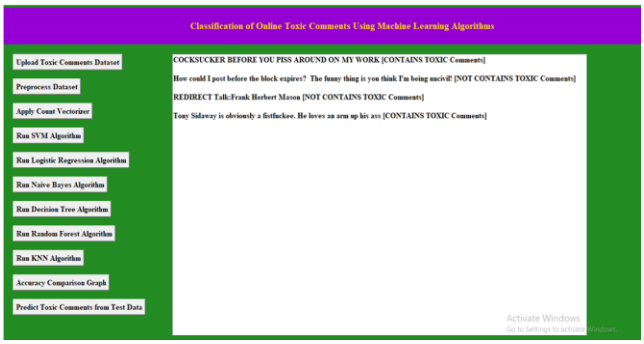
In above screen we can see accuracy and loss value for all algorithms and in above screen random forest gave 100% accuracy with 0% as loss and now click on 'Accuracy Comparison Graph' button to get below graph



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



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



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'

7. 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 ENHANCEMENT

Other machine learning models can be used to calculate accuracy, hamming loss, and log loss for better results. We

can also explore some deep learning algorithms such as LSTM (long short-term memory recurrent neural network), multi-layer perception, and GRU. So, we can explore many other techniques which will help us to improve the obtained result.

8. REFERENCES

[1] H. M. Saleem, K. P . Dillon, S. Benesch, and D. Rut hs, “A Web of Hat e: Trackling Hateful Speech in Online Social Spaces,” 2017, [Online] . Available: ht tp://arxiv.org/abs/1709.10159.

[2] K. K. Kumar, S. G. B. Kumar, S. G. R. Rao and S. S. J. Sydulu, "Safe and high secured ranked keyword search over an outsourced cloud data," 2017 International Conference on Inventive Computing and Informatics (ICICI), Coimbatore, India, 2017, pp. 20-25, doi: 10.1109/ICICI.2017.8365348.

[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,” Proc. 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] K. K. . Kommineni and A. . Prasad, “A Review on Privacy and Security Improvement Mechanisms in MANETs”, Int J Intell Syst Appl Eng, vol. 12, no. 2, pp. 90–99, Dec. 2023.

[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.

[9] E. Wulczyn, N. Thain, and L. Dixon, “ Ex machina: P ersonal at t acks seen at scale,” 26th Int. World Wide Web Conf. WWW 2017, pp. 1391– 1399, 2017, doi: 10.1145/3038912.3052591.

[10] H. Hosseini, S. Kannan, B. Zhang, and R. P oovendran, “ Deceiving Google’s P erspective API Built for Detecting Toxic Comment s,” 2017, [Online]. Available: ht tp://arxiv.org/abs/1702.08138.

[11] Y. Kim, “ Convolutional neural networks for sent ence classificat ion,” EMNLP 2014 - 2014 Conf. Empir. Methods Nat . Lang. Process. Proc. Conf., pp. 1746–1751, 2014, doi: 10.3115/v1/d14-1181.

[12] R. Johnson and T . Zhang, “ Effect ive use of word order for text cat egorization wit h convolutional neural networks,” NAACL HLT 2015 - 2015 Conf. North Am. Chapter Assoc. Comput. Linguist . Hum. Lang. Technol. Proc. Conf., no. 2011, pp. 103–112, 2015, doi: 10.3115/v1/n15-1011.

[13] Y. Chen and S. Zhu, “ Detecting Offensive Language in Social Media t o P rot ect Adolescent s,” [Online]. Available: ht tp://www.cse.psu.edu/~sxz16/papers/SocialCom2012.pdf.

[14] A. L. Sulke and A. S. Varude, “ Classificat ion of Online P ernicious Comment s using Machine Learning,” no. October, 2019.

[15] Kiran Kumar Kommineni, Ratna Babu Pilli, K. Tejaswi, P. Venkata Siva, Attention-based Bayesian inferential imagery captioning maker, Materials Today: Proceedings, 2023, , ISSN 2214-7853, https://doi.org/10.1016/j.matpr.2023.05.231.