Toxic Comments Classification using Neural Network

Rinal Patel, Hetal Gaudani

Abstract: ***Humans have built broad models of expressing their thoughts via several appliances. The internet has not only become a credible method for expressing one's thoughts, but is also rapidly becoming the single largest means of doing so. In this context, one area of focus is the study of negative online behaviors of users like, toxic comments that are threat, obscenity, insults and abuse. The task of identifying and removing toxic communication from public forums is critical. The undertaking of analyzing a large corpus of comments is infeasible for human moderators. Our approach is to use Natural Language Processing (NLP) techniques to provide an efficient and accurate tool to detect online toxicity. We apply TF-IDF feature extraction technique, Neural Network models to tackle a toxic comment classification problem with a labeled dataset from Wikipedia Talk Page.***

Keywords: Natural language processing, neural network, TF-IDF feature extraction, Toxic comments

I. INTRODUCTION

Today, people are able to express their opinions and discuss Today, people are able to express their opinions and discuss different aspects via social media platforms. In such a situation, it is quite obvious that argue may arise due to differences in opinion. But often these argues take a dirty side and may result in fights over the group during which offensive language termed as toxic comments may be used. These toxic comments may be threatening, obscene, insulting or identity-based hatred. So, these clearly pose the threat of abuse and harassment online.

Extreme negativities has sometimes stopped people from expressing themselves or made them give up looking for different opinions online [1]. According to a 2014 survey, 40% of Internet users were victims of online harassment [2]. The Conversation AI team, a research group founded by Jigsaw and Google have been working on tools and techniques for providing an environment for healthy communication [3]. They have also built publicly available models through the Perspective API on Comment Toxicity Detection [4]. But these models are sometimes prone to errors and do not provide the option to the users for choosing which type of toxicity, they are interested in finding. So, a more steady and flexible intelligent system is needed for Toxic Comment Prevention in social communication. The types of toxicity are simply toxic, severely toxic, obscene, threat, insult, and identity-based hate. The application is to overcome the drawback of the model developed using Perspective API, showing all the types of toxicity contained in the comment.

Revised Manuscript Received on May 20, 2020.

Rinal Patel, Department of Computer Engineering, G. H. Patel College Of Engineering, V V Nagar, Gujarat, India.

Prof Hetal Gaudani, Professor, Department of Computer Engineering, Gujarat Technological University, Gujarat, India.

II. RELATED WORK

Many Machine learning and Deep learning algorithms are used to detect types of toxicity in social media comments.

Julio C. S. Reis and Andre Correia proposed a knn, random forest, svm and naive bayes Approaches for text analytics in fake news detection, obtaining best Accuracy of 85% using Random Forest technique. In future they will take large volume of dataset and explore other techniques such as deep learning and push the boundaries of prediction performance.

Fahim Mohammad proposed a logistic regression,Bi-LSTM, XGBoost and naive bayes svm Approaches for text analytics in toxicity classification using n-gram feature extraction technique, obtaining best Accuracy of 80% using NBsvm and Bi-LSTM. In that, did not tune the parameters of different algorithms presented in there experiment.future work is to use word2vec / GloVe word embedding to see how they behave during the above transformations.

Peiman Barnaghi and John G. Breslin implemented a bayesian logistic regression and naive bayes with TF-IDF feature vector for opinion mining and sentiment polarity, obtaining best Accuracy of 74.84% using BLR machine learning technique. In future they will do trend detection relating to a topic on a set of streaming feeds, to determine the polarity of the target topics.

Chady Ben Hamida, Victoria Ge and Nolan Miranda applied various Deep Learning and Machine Learning approaches CNN, logistic regression and naive bayes for the task of detect toxic comments and find the bias, obtaining a Label Accuracy of 94.84% using CNN classification technique after extract feature via Glove embedding method.future work of particular paper is to improve preprocessing steps and would apply recurrent neural network with BLSTM.

Navoneel Chakrabarty et al. proposed a Machine Learning Approach involving Decision Tree Classifier with TF-IDF and bag-of-word feature generation technique for comment toxicity classification, obtaining a Mean Accuracy of 91.64% . In next the author can apply Grid Search Algorithm on the same dataset over the Machine Learning Algorithms

III. METHODOLOGY

This section will apply data pre-processing steps like cleaning data, remove stop words, and tokenize data on Wikipedia text dataset. Then extract the features using TF-IDF feature extraction technique.

This section mainly focuses on studying the effects of three different classification technique SVM, Decision Tree and Neural Network.

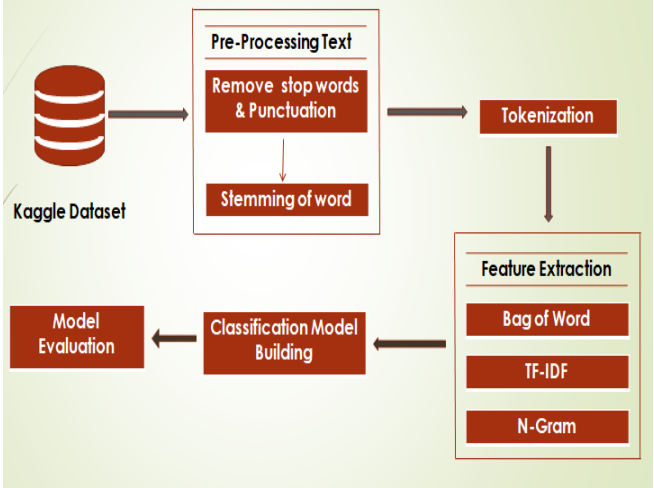

Fig. 1. Work flow diagram

A. Data Structure

The Wikipedia Talk Page Dataset prepared by Jigsaw and now publicly available at Kaggle is used [5]. The Dataset consists of total 159571 instances with comments and corresponding multiple binary labels: toxic, severe_toxic, obscene, threat, insult and identity_hate. Sample instances of the dataset are shown below in Fig, 2.

Table I. Structure of dataset

	id	comment_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
24696	414ace4068d42441	Also, it's well known that slavic women don't ...	1	0	1	0	1	1
13831	2481f24c79033e48	"\n\nI dismiss your RfC\n\nI dismiss your RfC as...	0	0	0	0	0	0
155289	bb9dc7a684501e50	I endure from your buddies and other ne'er do ...	0	0	0	0	0	0
11936	1fa2677e0e4628db	I thought the maximum has been 26% ethanol. I ...	0	0	0	0	0	0

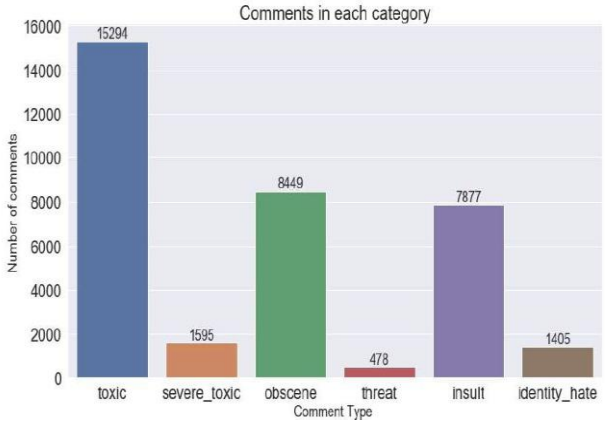

Fig. 2. Toxicity type

B. Data Pre-Processing and Feature Extraction

General text pre-processing steps are taken to convert raw text data into usable form for embedding model Training. In that first we clean unnecessary data like numbers, punctuation mark, extra space, articles, prepositions etc. Then stemming method used for remove affixes of word. After clean all things we need to tokenize data because these tokens are useful for finding such patterns.

Since most of the statistical algorithms, e.g. machine learning and deep learning techniques, work with numeric data, therefore we have to convert text into numbers. Several approaches exist in this regard. However, the most famous one is TF-IDF vector.

The TF-IDF value increases in proportion to the number of times a word appears in the document but is often offset by the frequency of the word in the corpus, which helps to adjust with respect to the fact that some words appear more frequently in general.TF-IDF use two statistical methods. In particular dataset number of generated feature using TF-IDF is 36410. For the training process of each model, we split the data for training and validation in the ratio 70:30, so that from 2000 data we had 1400 training points and 600 validation points.

C. Classification Model

▪ Support vector machine

We implemented a support vector machine classification technique with TF-IDF feature. SVM are one of the most powerful classification algorithms. The idea is to find an optimal hyper plane which divides the two classes accurately. There is also a concept of margin, which is the supposed to be maximum from both the classes so as to avoid any overlapping between two classes. Data which is not linearly separable is mapped into a higher dimension to achieve better classification results. Kernel functions such as radial basis function (rbf) and polynomial are used for non-linear data.
In case of toxicity detection, we used RBF kernel function with 1e-3 and 1e-4 gamma value.
Linear Support Vector Machine Algorithm:

1. The p-dimensional training instances (with p features) are assumed to be plotted in space.
2. A Hyperplane is predicted, which separates the different classes.
3. The best hyperplane should be selected finally, which maximizes the margin between data classes. The data points, influencing the hyperplane are known as Support Vectors.
4. The Large Margin Intuition for selection of best hyperplane for Linear SVM is given below:

$$\min C \sum_{i=1}^{m} [y^{(i)} \text{cost}_1 (\phi^T x^{(i)}) + (1 - y^{(i)}) \text{cost}_0 (x^{(i)})] + 1/2 \quad \sum(\phi_j)^2$$

Where, C is the penalty parameter, and ϕ is the parameter which needs to be optimized.

▪ Decision tree classifier

A decision tree is a tree-like graph with nodes representing the place where we pick an attribute and ask a question; edges represent the answers to the question; and the leaves represent the actual output or class label. They are used in non-linear decision making with simple linear decision surface.

After Tf-idf transformation, a complete numeric featured dataset is obtained. Now, a Decision Tree Classifier is instantiated.

Decision Tree Classifier Algorithm:

1. The best feature of the dataset is selected on the basis of Gini-Impurity and placed at the root of the tree.
2. The Training Samples are split into subsets such that each subset contains data with the same value for a feature.
3. Above two steps are repeated on all the subsets until leaf nodes are found in all the branches of the tree.

- **Neural Network**

Neural networks takes several inputs, process it through multiple neurons from multiple hidden layers and returns the result using an output layer. This result estimation process is technically known as "Forward Propagation". Next, we compare the result with actual output. The task is to make the output to neural network as close to actual (desired) output. Each of these neurons are contributing some error to final output. How do you reduce the error?

We try to minimize the value/ weight of neurons those are contributing more to the error and this happens while travelling back to the neurons of the neural network and finding where the error lies. This process is known as "Backward Propagation".

This one round of forward and back propagation iteration is known as one training iteration aka "Epoch".

In order to reduce number of iterations to minimize the error, the neural networks use a common algorithm known as "Gradient Descent", which helps to optimize the task quickly and efficiently.

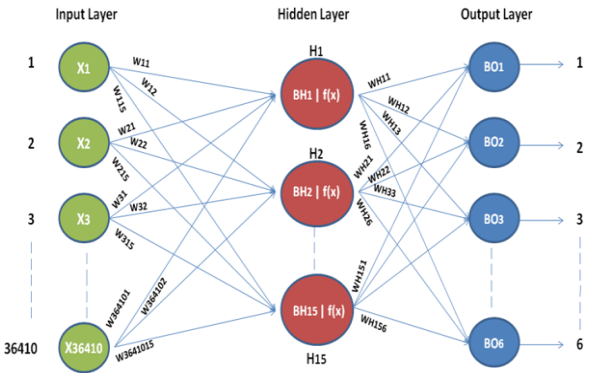

Fig. 3. Architecture of neural network

We implemented a neural network with back propagation as a classification technique. The inputs of the model are 36410 neurons. The neural network consists of 1 hidden layer with ReLu activation function, with 15 neurons. Since our goal is to perform multi-label classification, the output of the hidden layer is fed into a Softmax layer with 6 units, which correspond to the predicted probabilities of each of the 6 labels. The cross entropy function is used as the loss. The neural architecture can be defined as follows:

1. $h_1 = \text{ReLu}\ (xW_1 + b_1)$
2. $y_{predict} = \text{Softmax}(h_1W_2 + b_2)$
3. $J = CE(y, y_{predict}) = -\sum_{i=1}^{6} (y_i * \log(y_{predict(i)}))$

Where,

$x \in \mathbb{R}^{B \times 36410}$, $h_1 \in \mathbb{R}^{B \times 15}$, $y_{predict} \in \mathbb{R}^{B \times 6}$, $y \in \mathbb{R}^{B \times 6}$

And B is the batch size. The batch size is a number of samples or features processed before the model is updated .and the number of epochs is the number of complete passes through the training dataset. Here, we used 22 epochs.

IV. RESULT AND ANALYSIS

This toxic comment classification problem is multi-class as well as multi-label classification but svm and decision tree classification techniques are not supported both at the same time that's why we used six pipelines, each pipeline corresponds to each label. Using these pipelines, six models are built and trained separately.

Table II. Pipeline result for each label

	Validation Accuracy	
Pipeline/label	SVM	Decision Tree
1st pipeline/Toxic	91.66	93.16
2nd pipeline/Severe_Toxic	91.33	98.66
3rd pipeline/Obscene	91.83	96.50
4th pipeline/Threat	91.33	99.50
5th pipeline/Insult	91.66	94.66
6th pipeline/Identity_hate	91.33	99.16

Mean Validation Accuracy is the average of the Validation Accuracies achieved by the 6 Pipeline Models. Hence, it is the Mean Validation Accuracy of the 6 Headed Model prepared. From this model, a Mean Validation Accuracy is considered for svm and decision tree.

Table III. Comparative study of different classification techniques

Classification Techniques	Result (in %)
SVM	96.80
Decision Tree	96.94
Neural Network	97.07

V. CONCLUSION

This paper proposed a Machine Learning Approach combined with Natural Language Processing for toxicity detection and its type identification in user comments. In study we evaluate the accuracy of 97.07% by applying Tf-idf feature extraction method and Neural Network machine learning technique.

A more robust model can be developed by applying Recurrent Neural Network with long short term memory (LSTM) Algorithm on the same dataset over the Deep Learning Algorithms for multi label classification, being used in order to obtain better results and accurate classifications.

REFERENCES

1. CJ Adams and Lucas Dixon. Better discussions with imperfect models. url: https: / / medium . com / the - false - positive / better - discussions - with - imperfect - models-91558235d442.
2. Duggan, M., Rainie, L., Smith, A., Fuck, C., Lenhart, A., & Madden, M. (2014). Online harassment. Pew research center.
3. Conversation AI Team. https://conversationai.github.io/
4. Perspective API. https://perspectiveapi.com/#/
5. "Dataset" URL: https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data
6. CJ Adams and Lucas Dixon. Better discussions with imperfect models. URL: https://medium.com/the- false- positive/better-discussions- with- imperfect-models 91558235d442.
7. Reis, Julio CS, et al. "Supervised Learning for Fake News Detection." *IEEE Intelligent Systems* 34.2 (2019): 76-81.
8. Ying Liu, Han Tong Loh, and Aixin Sun. \Imbalanced text classification: A term weighting approach". In: Expert Systems with Applications (2009). URL: http : / /scholarbank.nus.edu.sg/handle/10635/60483.
9. Mohammad, Fahim. "Is preprocessing of text really worth your time for online comment classification?." *arXiv preprint arXiv:1806.02908* (2018).
10. Barnaghi, Peiman, Parsa Ghaffari, and John G. Breslin. "Opinion mining and sentiment polarity on twitter and correlation between events and sentiment." *2016 IEEE Second International Conference on Big Data Computing Service and Applications (BigDataService)*. IEEE, 2016.
11. Hamida, Chady Ben, Victoria Ge, and Nolan Miranda. "Toxic Comment Classification and Unintended Bias." (2019).
12. Jiang, Chuntao, et al. "Text classification using graph mining-based feature extraction." *Research and Development in Intelligent Systems XXVI*. Springer, London, 2010. 21-34.
13. Chakrabarty, Navoneel. "A Machine Learning Approach to Comment Toxicity Classification." *Computational Intelligence in Pattern Recognition*. Springer, Singapore, 2020. 183-193.
14. "Feed forward and back propagation technique for NLP" URL: https://www.guru99.com/backpropogation-neural-network.html
15. "Analyticsvidhya blog for neural network" URL : https://www.analyticsvidhya.com/blog/2017/05/neural-network-from-scratch-in python-and-r/
16. "Analyticsvidhya blog on Bag-of-Word feature extraction technique" URL:https://stackabuse.com/python-for-nlp-creating-bag-of-words-model-from-scratch/
17. Li, Yang, and Tao Yang. "Word embedding for understanding natural language: a survey." *Guide to Big Data Applications*. Springer, Cham, 2018. 83-104.
18. Liu, Bing. "Sentiment analysis and opinion mining." *Synthesis lectures on human language technologies* 5.1 (2012): 1-167.
19. Ibrahim, Mai, Marwan Torki, and Nagwa El-Makky. "Imbalanced Toxic Comments Classification Using Data Augmentation and Deep Learning." *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018.
20. Hinton, Geoffrey, et al. "Deep neural networks for acoustic modeling in speech recognition." *IEEE Signal processing magazine* 29 (2012).
21. "Deep learning approach to classifying types of toxicity in Wikipedia comments" Stanford education.URL: https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/reports/6838795.pdf.
22. "Detecting and Classifying Toxic Comments" Stanford education.URL:https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/reports/6837517.pdf

AUTHORS PROFILE

Rinal Patel, She Currently Pursuing A Postgraduate Degree Of Computer Engineering From G.H.Patel College Of Engineering At V V Nagar, Gujarat, India. She Has A Great Interest In Machine Learning, Deep Learning, Natural Language Processing And Computer Vision.

Prof Hetal Gaudani, She obtained Master's in Computer Engineering from Gujarat Technological University and Bachelor in Computer Engineering from Dharmsinh Desai University. Her main research interest includes Big Data Analytics, Machine Learning and Computer Vision. She has published many research papers in conferences and reputed journals. She has developed online IVR, GSM and biometric Based Hostel Management System for GCET Girls Hostel. She was Member of the Board of Studies in Computer Engineering in July 2011 in Sardar Patel University.