Vol. 10, Issue 10, October 2021

DOI: 10.17148/IJARCCE.2021.101009

Hate Content Detection and Removal from Online Social Network using Machine Learning

Kavya Patkar[1], Rutuja Yeole[2]

Student, Department of Computer Engineering, Sinhgad Institute of Technology, Lonavala, Maharashtra, India[1]

Student, Department of Computer Engineering, Sinhgad Institute of Technology, Lonavala, Maharashtra, India[2]

Abstract: In the last few years, the usage of online platforms has increased very rapidly. It lets the user to interact and upload content for others and also has become an integral part of many people's life. The user generated content posted by users contributes to the richness and variety of content on the web but isn't subject to the editorial controls associated with traditional media. Due to this some users can post content which could harm others, particularly children or vulnerable people. The amount of content generated on these platforms is increasing day by day, it's become impossible to spot and take away harmful content using traditional approaches at the speed and scale necessary. This paper examines the capabilities of machine learning technologies in meeting the challenges of moderating online content. For this we propose an Online Social Network System which does not allow its user to post hate comment or post. We have developed a machine-driven solution which automatically detects hate content in the form of text or image on social networking sites. For text classification various baseline models such as Support Vector Machine, Logistic Regression, Naive Bayes and Random Forest were used. The final model was a Support Vector Machine model that used TF-IDF for feature engineering. For image classification a combination of VGG19 pretrained on ImageNet dataset and BERT Language Model was used.

Keywords: Machine Learning, Natural Language Processing, Text Processing, Image Processing, Feature Extraction, Classification, Online Social Network, Hate, Non-Hate, Support Vector Machine, TF-IDF, VGG19, BERT

I. INTRODUCTION

Social network is interactive medium to communicate and share data related to human life. The data could be in the form of image, text, video and audio. It is a platform to build relationship among people who are interested in sharing views, pictures, real time connections and texts. Social network provides various types of services such as profiles, social links. Some of the social network sites which are used worldwide are Twitter, Facebook, Instagram.

Social Networking Sites are booming as never before. Currently social networking sites provide very less support to prevent hate messages on user wall. Apart from the numerous new opportunities that are provided, also hazards such as messages containing inappropriate content have to be taken into account and manually monitoring and analysing all messages separately is unattainable. This has resulted in the emergence of conflicts and hate, making online environment uninviting for users.

II. LITERATURE SURVEY

In [1], the paper provides a survey and state of the art natural language processing (NLP) technique that is used in automatic detection of the hate speech on OSNs, such as dictionaries, bag-of-words, N-gram etc.

In [2], the paper presents the numerous approaches based on traditional classifiers, deep neural models, and transfer learning models, along with features used for the classification. Results showed that the best classifier for the binary classification did not perform best in the multi-class classification, and the performance of the same classifier varies across the languages.

In [3], the paper proposes to model online conversations through graphs, and to perform the classification task using only graph-based features. They first extract a conversational network from raw chat logs and characterize it through topological measures. Then use these as features to train a classifier on abuse detection task.

In [4], the paper proposes an approach to detect hate expressions on Twitter. The approach is based on unigrams and patterns that are automatically collected from the training set. These patterns and unigrams are later used, among others, as features to train a machine learning algorithm.

DOI: 10.17148/IJARCCE.2021.101009

III. PROPOSED WORK

Our proposed model basically focuses on textual and image data. Whenever the user posts a comment or post in the form of text or image it is passed as an input to the classifier module where the comment or post is classified into hate and non-hate. If the data is hateful then it gets masked for all the users and the notification is send to user for whom the comment was posted with that exact hate message whereas if it is non hateful then it gets displayed as it is.

The main objectives are:

- To provide a healthy environment to the users.
- To avoid the display of hateful post and comments.
- Reduce toxicity in social media platform.
- Automatically detect hate content on social networking sites.

IV. SYSTEM ARCHITECTURE

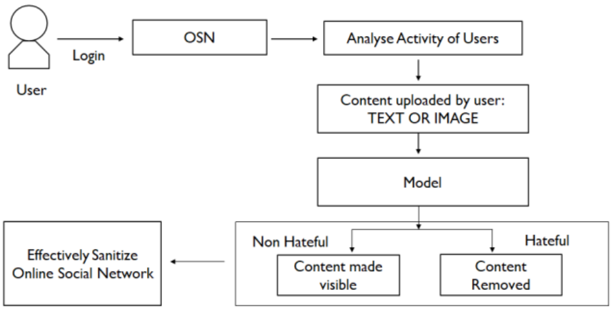

User
Login
OSN
Analyse Activity of Users
Content uploaded by user: TEXT OR IMAGE
Model
Non Hateful
Hateful
Content made visible
Content Removed
Effectively Sanitize Online Social Network

Fig. 1 System Architecture

A. Mathematical model

Let WS be the whole system which consists:

WS= {IP, PRO, OP}

Where,

1) IP is the input

IP= {U, S, OSN, SA, UA}
U is the number registered users in the system
S is the user settings
OSN is the system
SA is the sensitive attributes (Comments, Images and Post)
UA is the user activities

2) PRO is the procedure

The user will register with his/her basic information when he/she visits for the first time
Whenever he/she will comment on a post or post an image it will go through the classifier
If it is hateful, it will be not be posted and the user will get a notification but if it is not hateful then it will be posted

3) OP is the output

Online Social Network (OSN) free from hateful content

DOI: 10.17148/IJARCCE.2021.101009

V. IMPLEMENTATION

A. Text Classification Module

1) Dataset Description:

We have used the CrowdFlower dataset from data.world. The dataset is a .csv file with 24,784 text posts from Twitter where the tweets were labelled as hate speech, offensive and neither. The labels on this dataset were voted on by crowdsource and determined by majority-rules.

2) Pre-processing:

To prepare the data for binary classification, labels were manually replaced. The cleaned dataset consists of total_votes, hate_votes, non_hate_votes, label, tweet, clean_tweet. Cleaning process included the following tasks: reassigning labels, lowercasing tweet text, removing hashtags, mentions, quotes and punctuation from tweet text and checking for missing values followed by tokenization, removal of stop words and lemmatization. We found out that the corpus had a vocabulary of 20,277 unique words.

3) Feature Engineering:

The purpose of feature engineering is to transform the tokenized text data into numerical vectors that the machine learning algorithm can understand. In this experiment, we have used different feature engineering techniques like Count Vectorization, TF-IDF Vectorization.

4) Algorithms:

Support Vector Machine: Support Vector Machine model is basically a representation of different classes in a hyperplane in multidimensional space, the hyperplane will be generated in an iterative manner by SVM so that the error can be minimized.[5] The goal of SVM is to divide the datasets into classes to find a maximum marginal hyperplane (MMH).[5]

Logistic Regression: Logistic Regression is a supervised learning classification algorithm used to predict the probability of a target variable.[5] The nature of target or dependent variable is dichotomous, which means there would be only two possible classes.[5]
Naive Bayes: Naive Bayes algorithms is a classification technique based on applying Bayes' theorem with a strong assumption that all the predictors are independent to each other.[5] In simple words, the assumption is that the presence of a feature in a class is independent to the presence of any other feature in the same class.[5]
Random Forest: Random Forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. [5] It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.[5]

The best performing model was Support Vector Machine that used TFIDF for feature engineering.

B. Image Classification Module

1) Dataset Description:

The dataset for this experiment was created by scrapping data from Reddit. PRAW (Python Reddit API Wrapper) was used for this which is a python package that allows for simple access to Reddit's API. The data was downloaded from different sub-reddit pages according to the labels and ratings. The final dataset consists of 2740 images which were divided into three groups positive, negative and neutral.

2) Pre-processing:

We have used multimodal framework for this experiment. For this we need to extract the textual data from the images. We used Tesseract-OCR. Tesseract-OCR is Long Short Term Memory network based OCR engine which helps in detection and recognition of texts embedded in an image. This text is stored in a file and used further for processing.

For image pre-processing we first import the image followed by reshaping it to (224,224,3) as this this the input format for vgg19.These images are then converted into arrays and passed as input. The newly created file consisted of three columns namely image, filepath and text. After pre-processing we divide the dataset into training and validation set.

3) Algorithms:

VGG19: VGG is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper Very Deep Convolutional Networks for Large-Scale Image Recognition. The model achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes.[6] VGG19 is a variant of VGG model which in short consists of 19 layers (16 convolution layers, 3 Fully connected layer, 5 MaxPool layers and 1 SoftMax layer). There are other variants of VGG like VGG11, VGG16 and others.[7]

BERT: Bidirectional Encoder Representations from Transformers (BERT) is a Transformer-based machine learning technique for natural language processing (NLP) pre-training developed by Google.[8] The original English-language BERT has two models.[8] We have used the BERTBASE: 12 Encoders with 12 bidirectional self-attention heads for this project, this model is pre-trained from unlabeled data extracted from the BooksCorpus with 800M words and English Wikipedia with 2,500M words.[8]

We used a combination of VGG19 pretrained on ImageNet dataset and BERT Language Model. For text extraction we used BERT Language Model and for image extraction we used VGG19. We combined the output of these two models and made a concatenation layer. We passed this to two fully connected layer and then predict the final output in the last layer as positive, negative or neutral.

VI. RESULT

For this work, we mainly focused on F1-score along with precision and recall. The F1 score finds the harmonic mean between Precision and Recall, and it's useful for data with high class imbalance. The below chart depicts the performance of all the classifiers used.

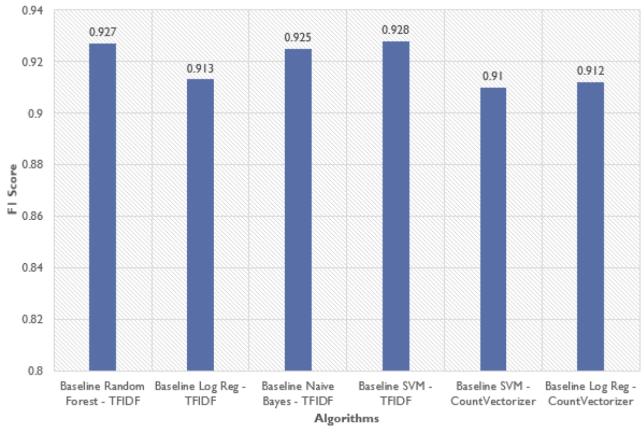


Fig. 2 Performance of all Classifiers

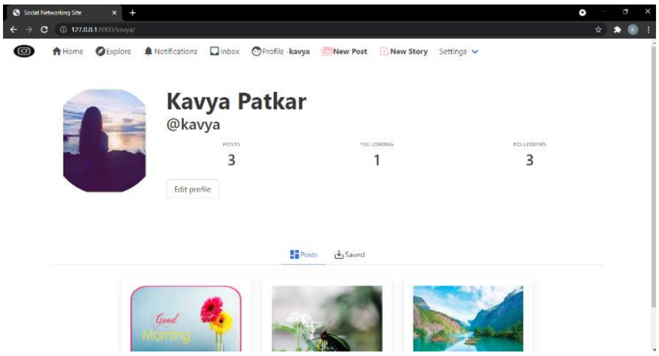
After comparing the performance of all the classifiers, the best performing classifier was Support Vector Machine that used TFIDF for feature engineering with a F1 score of 0.928. This model was used for final prediction for text classification. The image classification model which was a combination of VGG19 and BERT model gave an accuracy of 82.8%. This model was used for final image classification.

VII. USER INTERFACE

User Profile

Vol. 10, Issue 10, October 2021

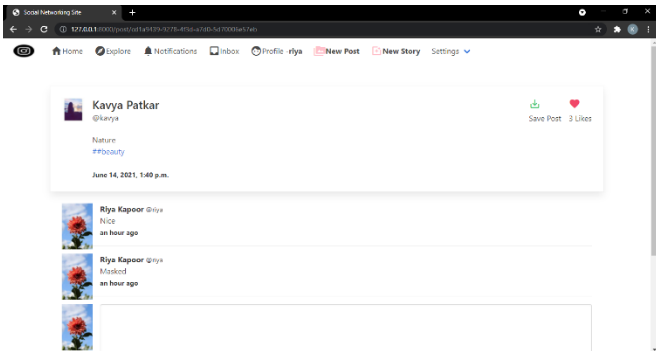
DOI: 10.17148/IJARCCE.2021.101009



Kavya Patkar
@kavya

POSTS	FOLLOWING	FOLLOWERS
3	1	3

Hateful comment Masked and Non-Hateful comment Displayed



Kavya Patkar
@kavya

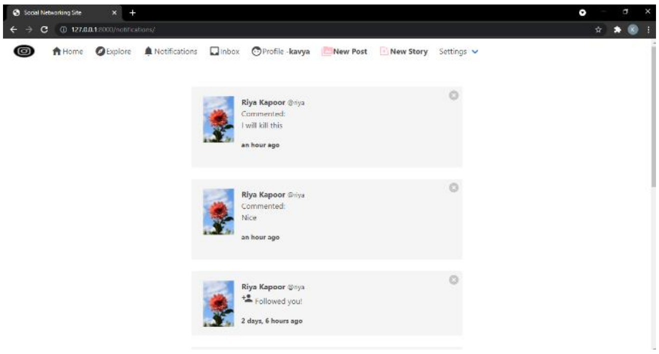
Nature
##beauty

June 14, 2021, 1:40 p.m.

Riya Kapoor @riya
Nice
an hour ago

Riya Kapoor @riya
Masked
an hour ago

Hateful comment displayed in user notification tab as it for whom the comment was posted but masked on post for all the other users



Riya Kapoor @riya
Commented:
I will kill this
an hour ago

Riya Kapoor @riya
Commented:
Nice
an hour ago

Riya Kapoor @riya
Followed you!
2 days, 6 hours ago

VIII. CONCLUSION

On studying about hate detection in online social network platforms we came to know that they use human moderators for hate content detection. But with our approach this can automated easily. In this paper, we implement an approach on hate content detection and removal from online social network based on images and text. This system can automatically detect post or comment containing hate content and does not allow the user to post it. Thus, provide a healthy environment to the users.

RFERENCES

[1]. Automatic Speech Detection on Social Media: A Brief Survey by Ahlam Alrehili ACS 16th International Conference on Computer Systems and Applications (AICCSA) IEEE 2019

[2]. Tracking Hate in Social Media: Evaluation, Challenges and Approaches Sandip Modha, Thomas Mandi, Prasenjit Majunder, Daksh Patel Springer Nature Singapore 2020

DOI: 10.17148/IJARCCE.2021.101009

[3]. Conversational Networks for Automatic Online Moderation Etinne Papegnies, Vincent labatut , Richard Dufour , and Georges Linares IEEE Transaction on Computation al Social System , 2019

[4]. Hate Speech on Twitter: A Pragmatic Approach to collect Hateful and offensive Expression and Perform Hate Speech Detection Hajime Watanabe, Mondher Bouazizi, Tomoaki Ohtsuk IEEE ACCESS 2018

[5]. https://www.tutorialspoint.com

[6]. https://www.cs.toronto.edu/~frossard/post/vgg16

[7]. https://iq.opengenus.org/vgg19-architecture

[8]. https://en.wikipedia.org/wiki/BERT_(language_model)