IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Prevention Of Cyber Troll And Sarcasm System On Social Networking Using Machine Learning With Bilingual Analytics

[1]Priya Harshkumar Mane, [2]Prof. Chirag Ramesh Desai, [3]Tejas Paresh Karia,

[4]Jeet Paresh Mehta, [5]Pratik Mahesh Merchant

[1]Student, [2]Professor, [3]Student, [4]Student, [5]Student
[1]Department Of Information Technology,
[1]K.J. Somaiya College of Engineering, Vidyavihar, India

Abstract: With the recent growth in use of social media, the trend of having an opinion for every content published on the internet has also increased. The internet allows users to remain anonymous in the sense that there is no compulsion for authenticating oneself, because of which many users exhibit indecent behavior by posting trolls, hate comments and spread negativity. The user who is the target of such actions may get seriously affected, he or she may slip into depression or lack of self-worth. To avoid such misconduct and essentially filter out such negativity, this paper proposes a web-application based deep learning solution to perform troll and sarcasm detection on comments received on a user's post and then using these comments, trace the user spreading hate and block or report him. The model consists of a Gated Recurrent Unit and a 4- layered neural network for troll detection and sarcasm detection respectively. As most of the comments posted in India are usually in Hinglish, which is a way of writing Hindi words using English letters, such comments will also be processed to determine the classification.

Index Terms **- sarcasm, troll, hinglish, deep learning, gated recurrent unit, social media.**

I. INTRODUCTION

This project will help to deal with online social media hate speech and automate the process of blocking such malicious accounts. The current process for the social media platforms are manual and there are no automated processes. Since the process is manual it becomes very difficult to keep track of such users who are habitual offenders. There are several categories of cyberhate and each of these are interpreted differently. The project has broken this down to mainly 2 categories: offensive and sarcastic depending on the sentiment and bilingual sentiment analysis on Hinglish comments. The main target for developing this tool is to empower influencers who do not have the time to tackle hate speech and thus they have to keep a social media manager who has to manually delete such malicious comments. Primarily, all the comments will be retrieved from the user created posts and then those will be classified using sentiment analysis. An automated response will be generated to the comments.

II. LITERATURE REVIEW

A. TROLL CLASSIFICATION

The work [1] in the research paper titled 'Incivility De- tection in Online Comments' co-authored by Farig Sadeque, Stephen Rains, Yotam Shmargad, Kate Kenski, Kevin Coe and Steven Bethard was referred to. Highlights from the paper are as follows: The dataset used was Russian troll dataset and the model was trained on newspaper comment data to detect any vulgar or offensive comments. A Recurrent Neural Network (RNN) architecture was used which had input, embedding, Gated Recurrent Unit (GRU) [2], Average pooling, Max pooling, Concatenation layers, and finally sigmoid activation function was used to make the binary classification of whether the comment was civil or not civil. The paper provides evidence that the model can also detect incivilities in Twitter or any other social media platforms.

B. SARCASM DETECTION

Various algorithms have been explored for detecting sarcasm in texts. The work [3] proposes a statistical approach for detecting sarcasm. The datasets from SemEval Task11 competition were used and statistical parameters were used to derive from tweets to help determine sarcasm in texts. Text cleaning was proposed and a set of 12 features were to be derived from the tweets for classifying tweets as sarcastic or not. These 12 features were divided into 2 categories - sentiment based features (word count for the positive and negative words, frequency of words with extreme positive or negative emotions, word count for nouns and verbs) and punctuation based features (count of repeated words, frequency count of dots, total count of question marks, exclamation marks, quotes and number of capital alphabets). Chi-square method was used for feature selection, from the array obtained through chi-squared method - p values were determined by sorting. The top 7 features were further selected based on the sorted p values list. For the initial approach, implementation of different machine learning algorithms like Decision Tree, SVM, KNN, Random Forest were done on selected features and the accuracy was compared with the accuracy that was obtained when all the features were used.

A few top features were identified after the feature selection was performed and those are:

- Positive word count
- Frequency of words with extreme positive emotions
- Frequency of words with extreme negative emotions
- Total number of verbs
- Count of nouns
- Repetitions of a letter
- Capital letter count

In the second approach, the total features with top 200 TF- IDF features were used. These features helped to determine a complex decision boundary. For better performance, ensemble models were used. The voting classifier was the learner method used to get maximum results among all the models and the final outcome is determined based on majority voting. In the first approach, the SVM algorithm gives an accuracy of 74.59% which is highest as compared to other algorithms. Similarly, for the second approach, the Voting Classifier model achieved an accuracy of 83.53%.

C. HINGLISH

The work [4] in the research paper titled 'Mixed Bilingual Social Media Analytics' co-authored by Saurabh Malgaonkar, Aejazul Khan and Abhishek Vichare was referred to. Following were some of the highlights of the paper: The dataset used was that of Twitter Corpus. Text Mining Techniques spanning across NLP was the approach followed. Extraction of data followed by cleaning of data followed by matching the keyword against the dictionaries such as English, Hindi and Hinglish were done. The algorithms used were Breen's algorithm and Cholesky decomposition for determining unknown sentiment. For better results, the Hindi dictionary should be well equipped with spelling ambiguities.

III. PROPOSED APPROACH

A. TROLL CLASSIFICATION

As comments or tweets are sequence data, Recurrent Neural Networks (RNN) work well with this type of dataset. The Gated Recurrent Unit (GRU) introduced in 2014 which is a type of Recurrent Neural Network has been used. It has been added in the model architecture as it adds a gating mechanism in addition to traditional Recurrent Neural Networks. Research shows Gated Recurrent Units show better performance on smaller or less frequent dataset. As few comments have offensive words at the start or beginning of the sentences, it won't be desirable that the neural network forgets them by the time it reaches the end of sentence. This is where the gating mechanism of Gated Recurrent Units comes into picture which enables the neural network to remember any offensive words at the beginning of a sentence by the time it reaches its end. Also to obtain word vectors the pretrained GloVe [5] model has been used. And to deal with high variance dropout layers[6] are used. Figure 1 shows complete flow of the troll detection algorithm.

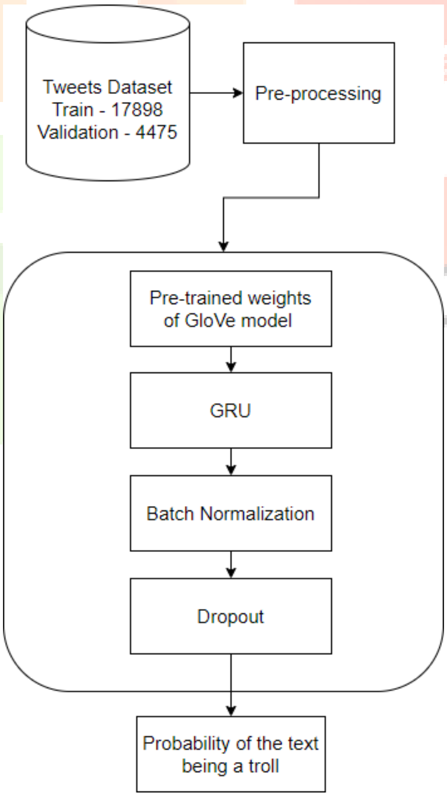

Fig. 1. Troll Classification Flow Chart

B. SARCASM DETECTION

The dataset used for sarcasm detection consists of 1984 tweets and it consists of 1024 Sarcastic tweets and 984 as non sarcastic tweets. The first step is basic text cleaning on tweets like converting tweet text to lowercase, removing urls, removing special characters and numbers, removing stop words. The second step is to split data into train (75%) and test (25%). The third step is to define a TensorFlow tokenizer to keep top 999 most frequent words only. It was also mentioned to replace out of vocabulary tokens with the ¡oov¿ tag. This tokenizer is fitted on a training tweets dataset. The texts to sequences method is used to transform each tweet in training sentences to a sequence of integers. It takes each word in the text and replaces it with its corresponding integer value from the word index dictionary. To make all the tweets of same length pad sequences is used. If a tweet is less than max length that is 100 length after pre-processing padding is added ”post” it and similarly if a tweet is greater than 100 length, letters after 100 count from the start are truncated. In the fourth step the four layer model is defined. Layer 1 is the text embedding layer. The output is a 2D vector of the embedding layer having one embedding there for each word in the input sequence of words. Layer 2 is a Flatten layer (GlobalAveragePooling1D), it flattens the 2D output matrix from the embedding layer to a 1D vector so that it can connect it to the next dense layer. Layer 3 is a dense layer with relu [7] activation function and with 24 units. The fourth layer is the final output layer with sigmoid activation function. Since the model is expected to return a probability of a tweet being sarcastic, a probabilistic loss is used for compiling the model i.e binary cross entropy. The model is then trained on the training data. An accuracy of 98.46% was achieved. Figure 2 shows complete flow of the sarcasm detection algorithm.

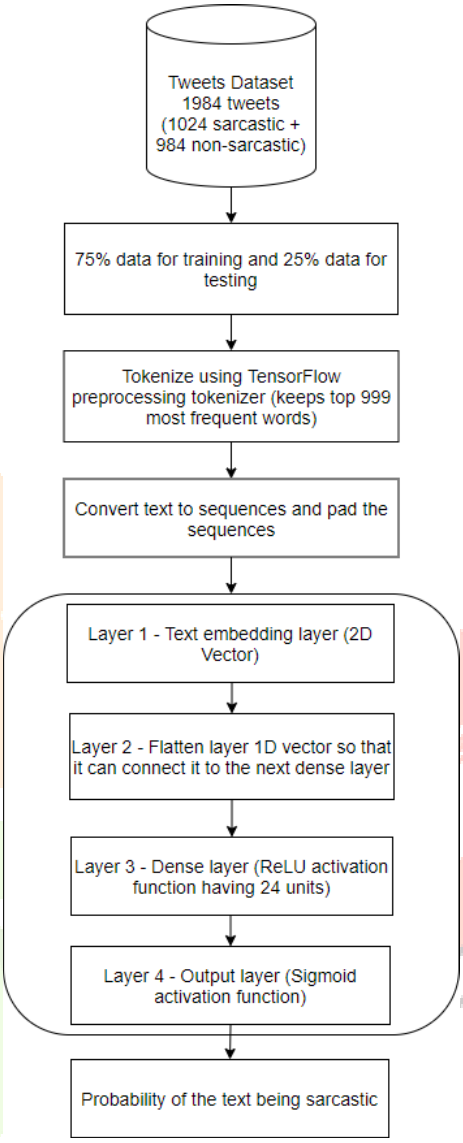
Tweets Dataset
1984 tweets
(1024 sarcastic +
984 non-sarcastic)
75% data for training and 25% data for testing
Tokenize using TensorFlow preprocessing tokenizer (keeps top 999 most frequent words)
Convert text to sequences and pad the sequences
Layer 1 - Text embedding layer (2D Vector)
Layer 2 - Flatten layer 1D vector so that it can connect it to the next dense layer
Layer 3 - Dense layer (ReLU activation function having 24 units)
Layer 4 - Output layer (Sigmoid activation function)
Probability of the text being sarcastic

Fig. 2. Sarcasm Detection Flow Chart

C. HINGLISH

As part of this project, Hinglish comments will also be classified. Hinglish is a combination of Hindi and English. This is largely used by speakers of Hindi which is usually constructed using Hindi vocabulary. After performing the pre-processing on the Hinglish comments, they will be fed as inputs to the algorithm which in turn will provide the classification for the same.

Figure 3 shows the complete system overview.

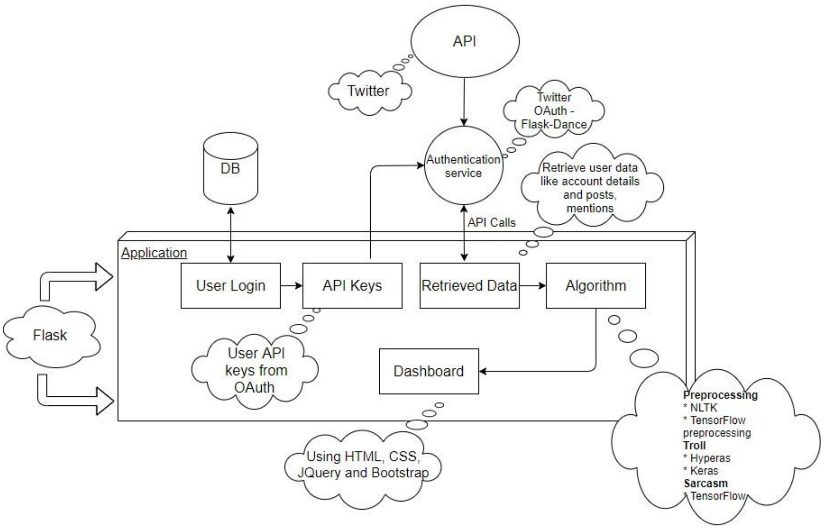

Fig. 3. System Overview

IV. RESULTS AND DISCUSSION

A. TROLL CLASSIFICATION

The model was trained and tested on tweets dataset containing tweet-id, text and label field where 1 depicts offensive tweet and 0 depicts not offensive tweet. The architecture uses pretrained weights from the GloVe model to get the word embeddings, Gated Recurrent Unit (GRU), and dropout layer to deal with high variance. The model is trained on 17898 samples and validated on 4475 samples. The accuracy of the model is 94%.

B. SARCASM DETECTION

The proposed model was tested on a tweets dataset. The tweets dataset was created by extracting tweets and analyzing the hashtags associated with each tweet. The tweets with hashtags like “#sarcastic” and “#not” were labelled as 1 for sarcasm and the others as 0.This data was split into training and test dataset in 3:1 ratio. The proposed model was trained on the training dataset. The model is a 4 layer neural network. The model has an accuracy of 98.46% and a binary cross entropy loss of 0.0662.

C. TROLL CLASSIFICATION

Sentiment analysis of Hinglish Twitter data was done using a pre-trained XLM-Roberta BERT [8] model. XLM-Roberta is a transformer based language model which relies on the Masked Language Model objective. This model was released by Facebook’s AI team as an update to their original XLM 100 model. XLM-Roberta was trained on a huge amount of training data which consists of the cleaned CommonCrawl data which takes up to 2.5 Terabyte of storage. The Ktrain library helped to build and train the neural network. The dataset used for training consists of Hinglish stop words, test labels, training set containing 14,000 tweets and development set containing 3,000 tweets. All these files are in txt format. The model has an accuracy of 37.73% as the volume of the dataset (number of records) is low.

V. CONCLUSION AND FUTURE WORK

The main aim here was to mitigate the spread of online hatred on social media platforms. The goal was to build a system that could filter the malicious comments and work hand-in-hand with the human moderators. This would help them reduce their workload and curb these kinds of activities. The technique used in Troll detection was with the help of Gated Recurrent Units and in sarcasm detection, a 4 layered neural network mechanism has been used. It was identified that there’s a limitation in sarcasm detection and it is difficult to make strong claims of the findings due to a small dataset. In spite of having this limitation, the model offered is quite efficient in detecting sarcasm in comparison to the other models that currently exist. Better results can be achieved if more work is done in collecting data and test the models again for consistent and efficient success.

VI. ACKNOWLEDGMENT

We are very pleased to present to one and all our research paper on Prevention of Cyber Troll & Sarcasm System on Social Networking using Machine Learning with Bilingual Analytics. We are very thankful to K.J. Somaiya College Of Engineering, Vidyavihar for supporting our ideas and giving us an opportunity to express them. We would also like to express our special gratitude towards Dr. Shubha Pandit, Principal, K.J Somaiya College of Engineering, Vidyavihar. We are also very thankful to our guide Mr. Chirag Desai, Professor, Information Technology Department, K.J. Somaiya College of Engineering, Vidyavihar for his continuous guidance and support throughout the project.

REFERENCES

[1] Sadeque, Farig, et al. ”Incivility Detection in Online Comments.” Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (* SEM 2019). 2019.

[2] Cho, Kyunghyun; van Merrienboer, Bart; Gulcehre, Caglar; Bahdanau, Dzmitry; Bougares, Fethi; Schwenk, Holger; Bengio, Yoshua (2014). "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation".

[3] R. Gupta, J. Kumar, H. Agrawal and Kunal, ”A Statistical Approach for Sarcasm Detection Using Twitter Data,” 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2020, pp. 633-638, doi: 10.1109/ICICCS48265.2020.9120917.

[4] Kaur, G.; Kaushik, A.; Sharma, S., “Cooking Is Creating Emotion: A Study on Hinglish Sentiments of Youtube Cookery Channels Using Semi-Supervised Approach” Big Data Cogn. Comput. 2019, 3, 37.

[5] J. Pennington, R. Socher, and C. Manning, ‘‘Glove: Global vectors for word representation’’, 2014, pp. 1532–1543.

[6] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov,”Dropout Layers:A Simple Way to Prevent Neural Networks From Overfitting”; 15(56):1929−1958, 2014.

[7] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” Haifa, 2010, pp. 807–814.

[8] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer and Veselin Stoyanov, “Unsupervised Cross-lingual Representation Learning at Scale”,2020.