Hate Speech And Abusive Language Suspect Identification And Report Generation

Shivdeep Chaudhari, Pooja Chaudhari, Pratik Fegade, Anand Kulkarni, Prof. Rahul Patil

Abstract: In recent years, there is a considerable amount of increase in distribution of hate speech on social media and thus some important action needs to be taken to reduce the criticism on social media. Despite an oversized range of rising scientific studies to deal with the matter, a serious limitation of existing work is that there is less efficiency as methods used were like TFIDF and stemming. This paper introduces a new method based on a deep neural network combining convolution i.e. Convolution Neural Network (CNN) and long short-term memory (LSTM) algorithm for classification of tweets into hateful or not. Based on the report generated by hate speech detection on particular topic we are going to track the users who are doing frequent hate speech, comments and tweets on social media platform.

Index Terms: hate speech, CNN, LSTM, term frequency inverse document frequency, N-Gram, porter stemmer, stemming, word2vec.

1. INTRODUCTION

Now days, as people are using various social media platforms like Twitter, Facebook, Instagram and other communication forum. Since last 15 years the numbers of users of social media platforms have been increasing tremendously. Thus, the huge amount of data is generated by interaction of people on such platforms. The statistics says that, the average daily time spent on social media by a person is 142 minutes a day. Every Second on an average around 60 tweets are tweeted on twitter. As these people can belong to various cultures these may lead to verbal assault and certain conflicts. According to Wikipedia Hate speech is defined as “any speech that attacks a person or group of people on the basis of attribute such as race, religion, ethnic origin, national origin, gender, disability, sexual orientation or gender identity”. We define offensive language as the text which uses abusive slurs or derogatory terms. Comments and tweets which are hateful are affecting people's life badly. Thus, it has become a necessity to detect the users who are frequently doing hate speech on social media and track them to take necessary action. Current scenario in such cases is that these things are done manually at FB, Twitter etc. It is consuming lots of human resources. In order to make it automated with high accuracy and more efficiency we are proposing our system is based on deep Neural network approach. As preprocessing plays a very important role in NLP. Hence, we are going to propose a model which is able to identify the hate speech and abusive language automatically tracks the user who is doing frequent hate speech. The report will be generated based on hate speech done by users. The system will take the most recent tweets from Twitter developer account. It will then preprocess the data by removing URL, emoticons, numbers. Stop words, the terns with very low frequency are also removed to avoid over fitting. Model will be trained on dataset from Kaggle which contains labels for hateful or not hateful". CNN + LSTM approach is used to build the model which will classify the tweet as hateful or not. The final report contains the users which are doing frequent hate speech.

2 PROPOSED SYSTEM

To overcome the issue which were introduced in the existing system and to increase the efficiency and accuracy. The system we are proposing as follows:

2.1 Preprocessing

At the time of Preprocessing the data need to be group in a manner. Hence there is a need to remove numbers, URLs, emoticon, non-ascii character, Negation words from the Dataset.

1. Remove number: Usually the numbers don’t give any idea about the speech is hateful or not. Hence, we remove the numbers from the dataset.
2. URLs: It seems that URLs don’t give any idea about the sentiment.
3. Emoticon: It doesn’t give much information regarding the tweets that it is hateful or not.
4. Non-Ascii: there are special characters like the words from German, Spanish language. These characters have generally no descriptive information in it. Hence, we don’t consider them.
5. Negation Words: The words like don’t, can’t have negation in it. So, we need to convert them into word like do not, cannot.

2.2 Stemming

The Porter Steamer algorithm is a process of removing suffixes from words in English. The terms with similar steam will usually have similar meaning. Test Word: READABLE Corpus: ABLE, APE, BEATABLE, FIABLE, READ, READABLE, READING, READS, RED, ROPE, RIPE.

Prefix	Successor Variety	Letters
R	3	E, I, O
RE	2	A, D
REA	1	D
READ	3	A, I, S
READA	1	B
READAB	1	L
READABL	1	E
READABLE	1	BLANK

2.3 N-GRAM

So, the output from the stemming will be given to N-gram. N-gram will convert the tokens into the group of N's, where N=1 means unigram, N=2 means bigram. In this model we will use bigram where n will be 2. For Example: If there is a tweet "This is not bad". So, if we don't apply bigram. It will separately consider "not" and "bad" then the sentiment value will be negative or so by considering n=2. we will consider "not bad" as a single entity and sentiment will be positive. N-gram helps to maintain sequential of data in which sequence of word matter to identify the meaning of a sentence.

2.4 Word Embedding

Word Embedding is collective name for a set of language modeling and feature learning techniques in n- gram where words or phrases from vocabulary are mapped to vectors or real number. In this method we are using GloVe, stands for Global Vector for word representation. It is an unsupervised learning algorithm resulting embedding from GloVe show linear substructure of words in vector space. The Glove results showcase nearest neighbor and substructure of words.

1. The Euclidean distance between two words vector represents similarity of corresponding words.

Eg: From, Frog, tadpoles etc.

Their Euclidean distance in Glove is nearly equal as they all are similar.

2. Linear Substructure: The similarity matrix produced a single vector that gives relatedness of two words. This is nothing but similarity and resemblance between two words.

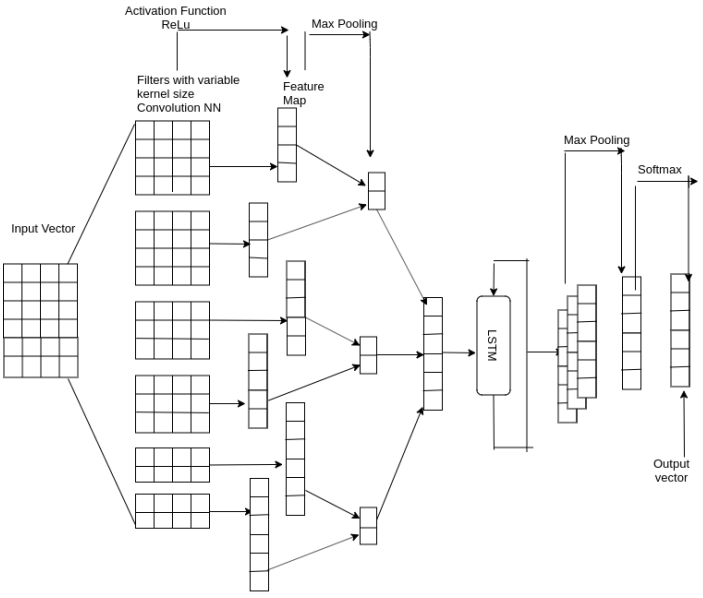
Eg: Students – college, school - principal, company – CEO etc. are linear substructure which shows similarity between two words.

Consider the following example,

Glove (India) – Glove (Australia) + Glove (Delhi) = Glove (Sydney).

3 CNN LSTM ARCHITECHTURE

The Architecture of the model is as shown in fig.1 At this stage the data obtained does not contain any emojis, symbol, numbers. The data is pre-processes using python library and classified using the algorithms like porter stemmer and n-gram. While passing pre-processes and classified data for further processing the sequence of the words in the sentence matters. The original sequence of the data should remain the same for the output. To keep the sequence of words same as it in the tweets. We will send this data to the convolution neural network. The sequence of the words in a sentence is remain same over the convolutional neural networks. At this stage the data is in a sequential format. this data Is passes onto the lstm for next processing. The convolutional layer consists of 100 nodes. These 100 nodes can be connected together. otherwise they can be used for getting connection between other nodes. Size of sentence can be varying according as per the description. Convolutional layer requires the data in same size so at each node the kernel size will vary in range of 2,3,4,5… and so on.

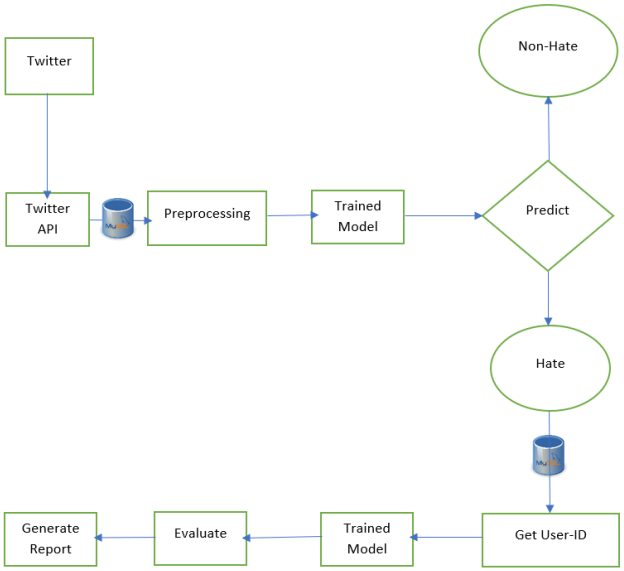


CNN + LSTM Architecture

So, when activation function will stimulate the neuron. we will get the feature map. Feature map contains the output matrix. The feature map will extract the patterns i.e. word sequence containing hate word, abusive language from the input tweets. The input tweets will be provided from the training dataset. For the more generous representation, Convolutional network model will apply set of filters. This set of filters will work in parallel so that generating multiple feature maps. This set of filters will generate a bank of filters which will sequentially evolved with sentence matrix and produce feature map matrixDecision boundaries can vary for word vector to word vector. To enable the learning of convolutional neural network for the different nonlinear boundaries we will apply activation function. the activation function will be relu among the different choices like sigmoid, hyperbolic tangent (tanh). The relu activation function will ensure that the feature map will always be generating positive. The reason behind relu, it will increase the accurate results for training dataset. Next pooling operation is applied on the output. The pooling operation can be performed using max-pooling or avg-pooling. Max-pooling is applied to the output generated from the activation function unit relu. At this stage the representation is minimized as far as possible. The main objective of max-pooling is to aggregate much more information as possible. In max-pooling rather than extracting single max values from the input data we can extract multiple values in their original input sequence as in the tweets . the max- pooling applied in the model will give the maximum value from the input. The largest value is obtained from the columns of feature map matrix by operating max-pooling on columns of feature map. We have combined this CNN and LSTM network for task of Hate Speech detection. This hybrid model has ability to learn word level features. Features which are extracted through CNN will be directly fed to forward LSTM Network as well as backward LSTM Network. A linear layer and a softmax layer will decode the outputs given by each LSTM layer into the probabilities for each category, and the vectors that will combine to produce the final output.

4 REPORT GENERATION

In accordance to the Twitter API rules and regulations, we define hate speech followed by the hateful user. Our hybrid CNN + LSTM based architecture would identify the tweet based on a particular hashtag and classify them as hateful or not. If the retweet of the hateful tweet is treated as a directed graph {V, E} we get a dense graph of the hateful users where, u1 is the starting vertex of an edge of one hateful user and u2 is the terminating edge of the another hateful user.As it is more likely that the hateful users will tweet hateful comment to another hateful users. Hence, a densely connected graph is obtained of hateful users. The users of these hateful tweets are identified from their user-id. We could also gain some information about the user"s current location, biographical information, number of twitter users being followed by the user, etc. In this way, a detailed report will be generated of the hateful users on the community.



3 CONCLUSION

Using various preprocessing methods and CNN + LSTM hybrid architecture, trained module will be able to identify users spreading hatred on trending topics. The final report will be generated which will specify users and their frequency of hate speech. We are planning to develop a system which will overcome limitations like symbols used in some abusive words or abbreviations having different meanings, these cases are reducing the accuracy. We are trying to find solutions for such cases.

REFERENCES

[1] Zhang, David Robinson, Jonathan Tepper, "Hate Speech Detection Using CNN-LSTM Based Deep Neural Network",Ziqi ACM The Web Conference WWW 2018, ACM New York.

[2] Zeerak Waseem University of Copenhagen Copenhagen, Denmark"Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter ".

[3] Aditya Gaydhani, Vikrant Doma, Shrikant Kendre, MIT "Detecting Hate Speech and Offensive Language on Twitter using Machine Learning: An N-Gram and TFIDF based Approach".

[4] Fabio Del Vigna, Andrea Cimino, Felice Dell"Orletta , Marinella Petrocchi , and Maurizio Tesconi"Hate me, hate me not: Hate speech detection on Facebook".

[5] Pinkesh Badjatiya , Shashank Gupta , Manish Gupta, Vasudeva Varma"Deep Learning for Hate Speech Detection in Tweets".

[6] Ona de Gibert Naiara Perez Aitor Garc ıa-Pablos Montse Cuadros HSLT Group at Vicomtech, Donostia/San Sebasti´an, Spain "Hate Speech Analysis from a White Supremacy Forum", by.

[7] Bjorn Ross Michael Rist Guillermo Carbonell ¨ Benjamin Cabrera Nils Kurowsky Michael Wojatzki "Measuring the Reliability of Hate Speech Annotations: The Case of the European Refugee Crisis ".

[8] Younghun Lee Seunghyun Yoon, Kyomin Jung "Comparative Studies of Detecting Abusive Language on Twitter".

[9] Thomas Davidson, Dana Warmsley, Michael Macy, Ingmar Weber"Automated Hate Speech Detection and the Problem of Offensive Language".

[10] Zeerak Waseem University of Copenhagen Copenhagen,"Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter", Denmark csp265@alumni.ku.dk Dirk Hovy, Dirk Hovy.

[11] Manoen Horta Ribeiro, Pedro Calais, Yuri Santos,"Characterizing and Detecting Hateful Users on Twitter", 12th International AAAI Conference on Web and Social Media (ICWSM 2018).