International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 07 | July 2020 www.irjet.net p-ISSN: 2395-0072

Social Media Harassment Analyzer

Nabeela Arfa[1]

[1]*Department of Computer Sci. & Engg, Thejus Engineering College, Vellarakkad, Thrissur, Kerala, India*

Abstract - *The strong rise in the use of various social media platforms has given rise to various anonymous exchanges of content in micro blogging platforms like Twitter. This is commonly termed as Cyber Harassment which makes use of email, messaging and websites to bully or harass an individual or groups through personal attacks like comments, offensive posts or tweets etc. Such kinds of abuse can bring about devastating effects on the victim's. Therefore an important goal is to detect such textual contents and consider it as a classification problem by integrating various techniques and tools to build a model that can detect and categorize them into their respective classes. It has also become a necessary condition to test these results in real settings and to report the abusive activities that are taking place currently in various social media platforms. This could be accomplished by building a deep learning model using RNN along with Bi-LSTM as an additional layer that can not only detect and classify the presence of Harassment languages but also help in finding solutions to report such intentional abuses by integrating the model with the twitter to collect real tweets.*

Key Words: RNN, Cyber Harassment, Bi-LSTM.

1. INTRODUCTION

Social Media harassment can be described as any violent, intentional action by individuals or groups in various social media platforms. This is commonly termed as cyber bullying or cyber harassment. Presence of such bullying content on social media sites has been creating devastating effects and hence it is considered as one of the top ethical issues today. It has been observed that such types of harassment or bullying occur when a bully approaches a person with hate and with an intention to harm his reputation. Nowadays, much of this bullying has been observed to be targeting youngsters and thus it has become common among teenagers. The bully approaches weaker sections of the society through personal attacks like comments, offensive posts or tweets. The term cyber harassment has become common as the bully's make use of emails, messages and websites to bully or harass an individual or group. The anonymous nature of such websites or messaging platforms has given rise to such unethical behaviors. Due to this nature of various social media platforms, there has been a growing interest by various researchers in detecting such tweets or contents that show the presence of abusive activities taking place currently in various social media platforms. Recent advances in Deep learning methodologies have motivated many researchers in considering such issues as classification problems by integrating various tools and techniques to build a model which can effectively mitigate this issue.

2. LITERATURE REVIEW

Several approaches that can detect cyber harassments and classify them into their types have been proposed so far. These approaches by different authors are discussed below. S. Salawu [1], in his work proposed a systematic review of various databases for the problem of harassment detection. Based on the results of previous approaches, his work deals with categorizing all the existing approaches into four main classes: supervised learning, lexicon based learning, rule based and mixed-initiative based approaches. He proposed the use of Supervised learning algorithms like SVM and Naive Bayes to build models fit for the purpose of cyber bullying detection.

R. Zhao [2], proposed a method for automatic detection of cyber bullying on social networks based on bullying features. This work focuses on developing a framework which is specific to cyber bullying detection. This approach tries to overcome the drawbacks of previous methods which just dealt with building text categorization models while ignoring bullying characteristics. For effective bullying detection, this approach makes use of word embeddings to convert the words into standard forms and to assign vectors to them.

D. Chatzakou [3], presented a robust methodology for the detection of aggression and bullying on twitter. This method uses a precise and scalable approach to extract text, user, and network-based attributes. It also studies various properties of cyber bullies and aggressors which is one of the main things that must be considered while detecting and solving this issue. It also concentrates on determining what features distinguish them from regular users and how we can classify them as either harassment or normal tweets. This method focuses on collecting datasets and building a ground truth. It also focuses on developing a model that can extract user based, text based and network based features.

Kshitiz Sahay [4], proposed a robust method for detecting cyber bullying and aggression in social networks using Natural language processing which extracts text, users and network based attributes and also studies the properties of bullies and aggressors and what features distinguishes them from regular users. This work explores various NLP and ML algorithms and evaluates them for the task of identifying bullying comments in a dataset. It mainly aims at exploring various possibilities of classifying hate speech, insults and

harassments which seem to be an extension of current and previous research works on cyber bullying and harassment detection. It has also made use of various feature engineering efforts to make use of the domain knowledge of data to create features that make ML algorithms work accurately.

T. Wohner [5], in his work of detecting online harassment in social networks has explored various pattern based approaches in order to detect harassment messages and also various normalization techniques that normalize each word in the text by transforming words in their canonical forms. This approach was adopted because of the presence of noisy language words in the user generated texts. It also deals with building a person specific module to mark phrases related to a person. The pattern based classifier which was then developed uses the information provided by various preprocessing steps in order to detect various patterns that would connect a person to certain swear words.

3. METHODOLOGY

For the development of a deep neural network model which holds the ability to classify various tweets as either harassment or not, the most important initial step is to collect the dataset and analyze all the dimensions which are necessary in building the model. The data in the dataset may be impure and may contain many noises or unwanted data which has to be first removed using various preprocessing steps like the data cleaning, data tokenization by using various word embeddings to convert them into their standard form, stop words removal, punctuation removal etc. After the extensive effort of preprocessing, the data in the dataset has to be divided and separated as training, test and validation sets. Some of the data or tweets are selected for the training purpose of the model while the remaining tweets are used up for the validation testing and overall testing of the detection model.

Next, the model that can predict the presence of harassment in tweets is being built using an RNN algorithm which is an extension of a CNN algorithm and can be used representing semantic relationships between temporal data like text. The main advantage of using RNN is that it contains many hidden layers which help in extracting all hidden information and features. An additional layer called Bi-LSTM or Bi Directional-Long Short Term memory can be integrated with RNN. Bi-LSTM allows learning over many steps as it learns long term dependencies very efficiently. Hence, this additional layer is mainly used for processing sequential data.

After the model is built, it has to be trained with appropriate data from the database and to check the validity of the model, a validation test will be performed using the validation test set that was taken during the database analysis. Finally the results can be tested with the test set to determine whether the tweets are harassment or not and then classify them into their respective classes. To test the results in real time and to enable real time data collection, the twitter can be integrated with the model. This will enable the analysis, processing and categorization of data in real time. The model will then give the results back to the twitter which will then allow the twitter administrators to take appropriate actions based on category.

Due to wide variety layers and their applications in Deep Neural Networks, some of the layers are integrated together to build the model. It can be divided into five layers.
(1) Input Layer
(2) Embedding Layer
(3) Dropout Layer
(4) Neural Architecture Layer
(5) Output Layer

3.1 Input Layer

This is the first layer in the architecture of the model to be constructed. The data that is collected from the dataset is given to this layer. This data is usually an input sequence which has been analyzed from the dataset and has undergone various preprocessing steps. This preprocessed input sequence is fed as an input to the next layer in the architecture.

3.2 Embedding Layer

The Embedding layer is used to represent each word as a vector of real valued words named as word embeddings vectors. This is an approach for representing words and documents using a dense vector representation. Classical NLP techniques fail to establish semantic relationships between words, hence GloVe, a recent word embedding technique that reconstructs linguistic contexts of words and establishes semantic relationships can be used for this purpose. This method transforms words into standard form which can then be sent to the neural network as a neural network can accurately classify the data only when these words are standardized.

3.3 Dropout Layer

The dropout layer is an additional layer in the neural networks which is added in order to avoid overfitting. Based on the requirement, any number of dropout layers can be added along with the neural architecture with different dropout rates. During the training process, neurons are selected and dropped randomly to avoid overfitting. This helps in improving the model performance as errors can be eliminated by backward weight updation process. In this process two dropout layers are added, that is one before the neural network and one after the neural network each of dropout rates 0.25 and 0.5 respectively.

3.4 Neural Architecture Layer

Recurrent Neural Network (RNN) is adopted as a neural layer to extract features from the input tweets. Bi-LSTM is used along with RNN as an additional layer. Bi-LSTM allows processing of various sequential data. The main advantage of using Bi-LSTM as an additional layer is that it learns long term dependencies between words and hence it allows the model to learn in depth over many steps. Bidirectional LSTM has been used instead of traditional and Unidirectional LSTM methods. The main advantage of using Bidirectional LSTM over other LSTM methods is that it allows the network to run in two directions. That is, from past to future and from future to past.

3.5 Output Layer

The output layer consists of two neurons to provide output in the form of output probabilities for each of the two classes' harassment and not harassment. This layer outputs the class of the given tweet and also categorizes tweets into their respective categories. This layer uses sigmoid function as an activation function to get a desired output which matches the target output.

4. IMPLEMENTATION

This section describes the implementation procedure carried out on the proposed methodology. The implementation is performed in five steps. In the following sections these steps are described in detail.

4.1 Data Preprocessing

The main aim of this step is to process the tweets that are collected from the dataset. The data that is collected from the database may not be in its pure or standard form. Hence, we need to perform various preprocessing steps on the dataset before giving it to the model to learn by its own. The Various steps that have to be performed under this module are dataset collection, data analysis, data cleaning, tokenization and data splitting.

For this approach, a cyberbullying dataset was collected from a live competition at kaggle. This dataset was originally created to identify and classify toxic online comments. In general it contains 95981 samples of comments along with their respective labels. These comments were collected from Wikipedia. The labels that were present in the dataset were classified as Toxic, Obscene, Threat, Insult and Identity Hate. From this dataset 80% of the data was used for the training process and 20% was used for the testing process, from which 10% was used as a validation test set while another 10% was used as output test set.

In data analysis the tweets that are collected from the dataset are analyzed. All the dimensions are analyzed and identified from the dataset. This step is very essential in order to build up an understanding of various relationships among the terms mentioned with their dimensions in the dataset.

The data in the dataset may be in its impure form as it may contain many unwanted or irrelevant features or terms. These irrelevant terms are hence called noise. Hence, it is very necessary to remove those terms before we feed the data into the model for training. This is called feature extraction as all the irrelevant features are removed and ignored while only the relevant features are extracted from the dataset.

The data tokenization step consists of representing each word as a vector of real values known as word embeddings vectors. These word embeddings allow us to compute the distributed representations of words in the form of continuous vectors. Word2Vec can be used for explicitly encoding many semantic relationships in addition to the linguistic regularities and patterns into the new embedding space.

The dataset has to be split finally for training and testing the model. A higher fraction of data is selected from the dataset for the training purpose and the remaining portion of the dataset for testing the data. The training split will be chosen again to split into two parts: a small fraction becomes the validation set with which the model is evaluated and the rest is used to train the model.

4.2 Model

A model that can predict and detect whether a tweet is harassment or not and that can classify those tweets into their respective harassment categories has to be constructed. This model can be used with a special type of convolutional neural network called the recurrent neural network (RNN). An additional hidden Layer called Bi-LSTM can also be incorporated while building the model and all the layers mentioned in the above architecture can be added as such. Finally, a sigmoid function can be used as an activation function to get a desired output which matches the target output.

4.3 Training and Testing

After the harassment detection model is built, the RNN model can then be trained with the training data from the training dataset. The validity of the model has to be performed next using the validation data along with the training of the model to determine the training accuracy. Finally, the fully built model can be tested using some test data from the test dataset.

4.4 Twitter API

As an extension to the fully built model, it is possible to integrate twitter with the model using various twitter API's provided by twitter. This mechanism enables us to collect real time data or tweets from twitter using the API's. The collected tweets are directly fed into the model as inputs for the analysis and detection of the presence of harassment. The trained model can then easily perform an analysis on those tweets and return a result of whether a tweet is harassment or not and also its respective category back to the twitter using an appropriate API. This will then help twitter to take actions based on each category.

After the module is trained, the social media analyzer collects tweets from twitter using a GetTweets function. The twitter then returns a json file containing the tweets and details of those users who posted them. This data is then sent to the trained model for the prediction and categorization of tweets. The predicted data is then sent back to the analyzer which then analyzes which harassment classified tweet corresponds to which user and finally initiates functions like Report(user) or Block(user) to twitter requesting it to block certain users whose post was classified as harassment.

5. CONCLUSIONS

A method for detecting and classification of harassments in tweets is proposed. The method proposed uses a combination of two deep learning architectures RNN and LSTM to classify tweets as harassment or not harassment. The Recurrent neural network architecture is modified to work with the twitter data set provided by Twitter. Harassment detection and classification can be performed by applying recent deep learning text pre-processing techniques like word embeddings to convert the words in the tweets to their standard form. The classified results can be tested in real time using an analyzer in the backend. Once verified and classified, these results are used to report the users who posted those comments. These users can also be blocked based on the severity of the harassment. Our results are comparable to traditional machine learning algorithms while using a significantly lower number of parameters and greater accuracy. A deep neural networks ability to handle large amounts of complex data has proved its effectiveness in solving various issues like cyber harassment detection. Its sophisticated architecture effectiveness showed that it outperforms various other classification methods. It has increased scalability and flexibility in accommodating various changes to target various other social media platforms. These solutions have brought greater relief to the social media users as those social media platforms are frequently visited by them. As these solutions can classify tweets and display their severity with greater accuracy, various other problems can also be integrated with this existing method, thereby providing greater flexibility.

REFERENCES

[1] S. Salawu, Y. He, J. Lumsden, "Approaches to automated detection of cyberbullying: A survey", IEEE Transactions on Affective Computing, vol. 11, Oct 2017.

[2] R. Zhao, A. Zhou, K. Mao, "Automatic detection of cyberbullying on social networks based on bullying features", Proceedings of the 17th international conference on distributed computing and networking. ACM, 2016.

[3] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, A. Vakali, "Mean birds: Detecting aggression and bullying on twitter", Proceedings of the 2017 ACM on Web Science Conference. ACM, pp. 1702-0687, 2017.

[4] K. Shitiz Sahay, "Detecting Cyberbullying", International Journal of Engineering Technology, Science and Research, ISSN 2394-3386, vol. 5, January 2018.

[5] U. Bretschneider, T. Wöhner, R. Peters, "Detecting online harassment in social networks", ISSN 2394 – 3386, vol 5, 2016.