Toxic Comment Classification Using Convolutional Neural Network

Under Guidance of Mr M. Venkata Krishna Rao

M. Sree pooja[1] Student

M. Mounika[2] Student

K. KavyaSree[3] Student

Y. Akhil [4] Student

Computer Science and Engineering Department, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India.

***Index Terms* - Conventional Neural Network, Word Embeddings, Stop words.**

Abstract — Today's generation spends most of their time in social networking applications or websites, people share their life experiences, express their feelings and their lifestyle by posting photos, stories, Live Streaming, profile pictures etc.

There is freedom for everyone to post the comments on the social network sites that may hurt people feelings and may disturb their life too which may cause huge loss.

The freedom for people to comment on any topics or words without any restriction has become an issue and this is extremely been affected many lives in cases as cyber bullying, harassment.

The harassment physically or direct is controlled by police and other forces but online harassment should be controlled by some models that restrict the user not to post a comment by identifying the comment toxicity level.

In this paper, the system defined a machine learning model that identifies the comment toxic type and level of toxicity for each toxic type. The proposed model is defined using CNN I.e. convolutional neural network.

I. INTRODUCTION

In recent days youth has been using social media websites for sharing a lot of information and social interaction has been widely increased on the internet. As social media has been giving freedom to share people's expressions and their lifestyle. Unfortunately, this freedom has become an issue with some people by posting inappropriate comments to other posts and this is leading to so many people facing depression and even falling into suicide situations.

Cyber bullying and online harassment have become major issues on social media which are affecting so many lives.

To overcome these issues a multi-labeled classification model is developed that classifies the level of toxicity of each comment like identity-based hate, threat, insult, and obscene. The datasets used in the model are taken from Wikipedia talk page edits.

The datasets contain three hundred thousand data rows collectively for training and test datasets which helps the model to classify the comment more accurately.

Convolutional neural networks (CNN) has been widely used for Image recognition or Image Processing in computer vision, it is core most model used in computer vision. Recently convolutional neural networks (CNN) have been interestingly used in NLP and have gained quite good and accurate results.

The main steps involved in the system for classifying a given comment are building the model using Word embedding CNN and validating the model then detecting the type, level of toxicity for comment by generating pickle File. The trained model generates the probability of toxicity of any sentence/comment for each toxic label that passed as input to the model.

Convolutional Neural Networks (CNN) has been widely applied for classification problems in different fields like text classification. In [3], Kim used CNN's to address a series of sentence-level classification tasks.

The proposed system is defined or can be formulated as two steps, first, we define the model then validate the model using pickle file by user-generated inputs.

II. RELATED WORK

Text classification is the main problem in NLP, In order to achieve good accuracy many machine learning models have been used like binary classification using regression techniques, KNN classification model in multiple research projects.

Nagwa El-Makky's [1] binary toxic/non-toxic classifier predicts if the input comment is toxic or not and a multi-label classifier that detects the different types of toxicity in toxic comments.

Recently Georgakopoulos et al.[2] have tackled the problem Toxic comment classification with Wikipedia talk page edits using CNN but he has proposed a system that classifies whether a comment is toxic or not. They do not address the problem of finding the types of toxicity present in the comment. Their solution also avoids the problem of data classes' classification by using a balanced subset of the data for building the model.

III. IMPLEMENTATION

The proposed system has built a deep learning model that classifies the user input comment or sentence into toxic labels with a level of toxicity by finding the probability. To find these probabilities we have used Convolutional Neural Networks which contains several layers which detect the probability for each label.

In this section, we discuss the steps involved in the proposed system and about the models that have been used in each stage.

A. Data Acquisition

For model training and evaluation the training, test datasets are taken from Wikipedia talk page edits the dataset contains comment id, and six toxic type labels like toxic, severe_toxic, obscene, threat, insult and identity_hate.

The datasets contain three hundred thousand data rows collectively with train and test datasets. 70% of the data has been used for training and 30% of data has been used for testing the model.

B. *Data Preprocessing*

Major steps involved in this stage are data cleaning, Standardization of data and tokenization. In online user-generated or input data contains several spelling mistakes, unnecessary data, symbols and links in comments that people post, hence the data contains several unnecessary symbols that should be cleaned. In data cleaning phase the train and test datasets are cleaned by replacing Non-valid strings with null strings, the input comment length to be exactly N tokens by padding shorter comments with a dummy word and truncating longer ones.

By analyzing the comments Toxicity classes distribution in training set with and without performing different data classification methods. Lengths in the training dataset set, we chose N = 150 words since most of the comments have less than or equal to 150 words. Now the data is standardized by replacing symbols like ! @,#,$,^,&, https etc and convert the data to a unique case.

NLTK is also known as Natural Language Toolkit is been used for tokenization of data into tokens. This tokenizer divides a string into substrings by splitting on the specified string into tokens

After data cleaning tokenizing is applied on data frames then tokens are generated, these tokens are converted into vectors using word2vector model. The main steps involved in data processing are Preparation for removal of punctuation marks and other symbols, Updating the list of stop words and removing stop words using NLP and Applying Countvectorizer to convert tokens into vectors using word2vector models.

C. Model Implementation

After data preprocessing the word embeddings is applied to both train and test datasets, apply() is used to apply tokenization for each row of the comment_text column.

Now define the CNN model using train embeddings and embeddings dimensions of each vector of train data, CNN has been widely applied on image processing in the era of computer vision but recently it has been applied on text classification and got some interesting results. Hence it has been used to develop deep learning model in proposed system the CNN consists of several layers they are Convolution Layers, consists of a number of Kernel matrices that perform on their input associate degree turn out an output matrix of options wherever a bias price is value-added. The learning procedures aim to coach the kernel weights and biases as shared nerve cell affiliation weights.

In Stage II Pooling Layers are integral parts of the CNN. The purpose of a pooling layer is to perform spatial property reduction of the input feature pictures. Pooling layers make a sub sampling to the output of the convolutional layer matrices combing neighboring elements. The most common pooling function is the max-pooling function, which takes the maximum value of the local neighborhoods. In Stage III Embedding Layer could be a special element of the CNN's for text classification issues.

Here, every word of a text document is reworked into a dense vector of fastened size.

In Stage IV Fully-Connected Layer is a classic Feed-Forward Neural Network (FNN) hidden layer. It is often understood as a special case of the convolution layer with kernel size one × one. This type of layer belongs to the class of trainable layer weights and it is used in the final stages of CNN.

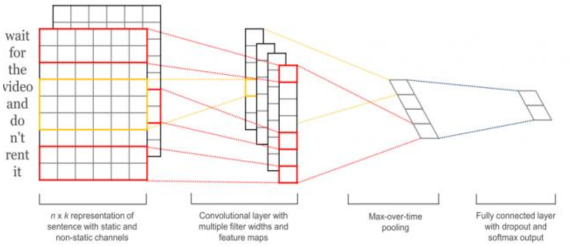
wait
for
the
video
and
do
n't
rent
it
n x k representation of sentence with static and non-static channels
Convolutional layer with multiple filter widths and feature maps
Max-over-time pooling
Fully connected layer with dropout and softmax output

Fig 1.Convolutional Neural Network for Sentence Classification

In Fig 1, CNN model is defined using word embedding of train data using ConvNet method. The embedded data is sent to convolution layers of a neural network, and then the output is sent to pooling layers where max pooling is applied. The outputs in each layer are formulated using ReLU (Rectified Linear Unit) activation function for hidden layers. The ReLU activation function can be used only for hidden layers and is defined as below :

$$\begin{aligned} \mathbf{R(X)} &= \mathbf{max(0,X)} \\ &= \mathbf{0} \quad \mathbf{if\ X<0\ and} \\ &= \mathbf{X} \quad \mathbf{if\ X>=0} \end{aligned}$$

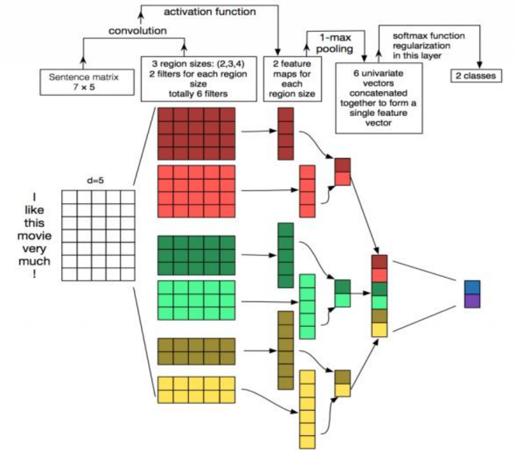
activation function
convolution
Sentence matrix 7 × 5
3 region sizes: (2,3,4) 2 filters for each region size totally 6 filters
2 feature maps for each region size
1-max pooling
6 univariate vectors concatenated together to form a single feature vector
softmax function regularization in this layer
2 classes
d=5
I
like
this
movie
very
much
!

Fig 2. CNN with Word Embeddings

The output of the hidden layer is feed into a sigmoid layer I.e. used to achieve binary classification (1, 0) for every 6 labels and squash the output between the bounds of 0 and 1.

"In Fig 2, the diagram depicts three filter region sizes: 2, 3 and 4, each of which has two filters. Every filter performs convolutions on the sentencing matrix and defines (variable-length) feature maps. Then 1-max pooling is performed over each map using max pooling, i.e., the largest number from each feature map is recorded. Thus a unique feature vector is generated from all six maps, and these 6 features are concatenated to form a feature vector for the penultimate layer. The final softmax layer then receives this feature vector as input and uses it to classify the sentence; here we have a tendency to assume binary classification and therefore depict 2 doable output states."

IV. MODEL VALIDATION

The model built in system will be able to predict the probability or level of toxicity for each toxicity type toxic, severe_toxic, obscene, threat, Insult, identity_hate. To validate the model a pickle file is generated and the model is loaded into file to detect the level of toxicity for each toxic type for the comment entered by the user.

V. CONCLUSION

The paper mainly addresses the abusive statements in social networking sites, uses natural language processing and CNN to detect and classify inappropriate text. The model developed is a multi-labeled classification scheme that can predict the different types of toxicity and levels of toxicity in a comment. This model basically calculates the probability of toxicity of comment or sentence, hence when it is implemented in any social network site the probability can be checked so that restriction for posting the comment is effective.

VI. REFERENCES

[1] Rahat Ibn Rafiq, Homa Hosseinmardi, Richard Han, Qin Lv, and Shivakant Mishra. 2018. Scalable and Timely Detection of Cyber bullying in Online Social Networks. In SAC 2018: SAC 2018: Symposium on Applied Computing , April 9–13, 2018

[2] Georgakopoulos, Spiros V., et al. ”Convolutional Neural Networks for Toxic Comment Classification.” arXiv preprint arXiv:1802.09957 (2018)

[3] Kim, Yoon. ”Convolutional neural networks for sentence classification.” arXiv preprint arXiv:1408.5882 (2014)

[4] Kaggle, “Toxic comment classification challenge,”2018.https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/leaderboard.

[5] Tokunaga RS. Following You Home from School: A Critical Review and Synthesis of Research on Cyber bullying Victimization. Computers in Human Behavior. 2010;

[6] Cowie H. Cyber bullying and its impact on young people’s emotional health and well-being. The Psychiatrist. 2013

[7] Joulin, Armand, et al.”Fast text. Zip: Compressing text classification models.” arXiv preprint arXiv: 1612.03651 (2016).

[8] Wang, Sida, and Christopher D. Manning. ”Baselines and bigrams: Simple, good sentiment and toxic classification.” Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers- Volume 2. Association for Computational Linguistics, 2012.