2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS) | 978-1-6654-0816-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICACCS54159.2022.9785067

Classification of Toxicity in Comments using NLPand LSTM

[1]Anusha Garlapati, [2]Neeraj Malisetty, [3]Gayathri Narayanan
[1,2,3] Dept of Electronics and Communication Engineering,
Amrita Vishwa Vidyapeetham, Amritapuri, India
garlapatianusha@am.students.amrita.edu[1],neerajmalisetty@am.students.amrita.edu[2], gayathrin@amrita.edu[3]

Abstract – With the increased usage of online social media platforms, there has been a sharp hike in toxic comments. Toxicity must be reduced. Classification of toxicity in comments has been an effective research field with various newly proposed approaches. This research and analysis provide a novel usage of the Natural Language Processing approach to classify the type of toxicity in comments. This analysis intends to interpret the type of comment and determine the various types of toxic classes such as obscene, identity hate, threat, toxic, insult, severe toxic. The input to our algorithm is comments from online platforms like toxic or non-toxic. Our model aims to predict the toxicity class.This project intends to analyze in phases. In Phase I, the objective is to evaluate the toxicity in comments by giving data through various techniques like TDIDF, spacy that helps data to perceive how every word in a comment is classified into a particular category of toxic class. Here, Algorithm will take comments from test data and predict the type of toxicity for test data like a toxic, threat, and so on. In Phase II, Data is analyzed to organize the comments into toxic and non-toxic categories. This promotes us to perceive the particular comment is toxic or not.

Keywords - Toxicity, Comments, NLP, TFIDF, Severe Toxic, Insult, Text Classification

I. INTRODUCTION

Increasing in usage of online platforms that also permits people to get across with each other, by exchanging feelings or attitudes about several events and they leave comments about their opinion. This has contributed to the evolution of Natural Language Processing (NLP). The background for the problem statement arises from the crowd of online comments. Identifying the toxicity in [1] comments has been a tremendous challenge for users. In a few cases, these online conversationsinclude certain language that can be divided into different types like toxic, insult, obscenity, severe toxic, threat, identityhate. Here, data has multi-labels to classify, each having binary classifications (0 and 1). So, the problem statement is considered to be a Multi-label classification problem. Unintended bias in these online conversations must be decreased. Although social media offers a lot of positive newsto world it also has negative aspects.

Toxic comments are examined like comments that are insulting, disrespectful which results in people leaving the conversations. Toxicity is like an argument that can point to that people both to stop honestly communicating themselvesand to stop seeking others' points of view out of fear of indignity. Automatic examining of toxicity in comments on social media platforms is helpful for people on social media users who could get warnings or unwanted messages [10]. Natural Language Processing with efficient algorithms is the most influential tool that is useful for users to analyze and classify several parameters from text-based comments by extracting the data. The aim was to develop a model that canpredict to classify the types of comments.

II. LITERATURE REVIEW

Various Machine learning and Deep learning methods are used for classifying the type of toxicity by using neural networks. There are various papers on toxicity classification [2] but they have used the constant model to predict accuracy. Conversational AI Team, that's a research initiative that was established by Jigsaw and Google to protect the conversation voices by building a technology [9]. It's easy for humans to divide or classify the text or images but it's difficult for computers as it deals with only binary values or numbers. Any data must be converted to a numeric type before applying it to the model. Classification of text uses NLP and various machine learning methods to classify comments.

Here, the intention is to explore the visualization of comments in different ways by utilizing some unique visualization techniques. LSTM and GRU are algorithms used for classifying the type of toxicity and prediction of test data in comments. Our analysis is to enhance online conversation by inspecting various techniques in NLP to create an authentic model that is adequate for detecting types of comments such as toxic or non-toxic and that model will be deployed using Heroku.

III. DATA SET

The data set consists of nearly 109449 rows and 8 columns such as comment text, insult, id, toxic, insult, identity hate, obscene, severe toxic. Each class in the data set has binary values 0 and 1, inferring whether a particular comment is related to that particular class or not. Table 1 and 2 The sample of data including rows and columns is as follows:

TABLE1: Sample Data

Comment Text	Toxic	Severe Toxic	Obscene
That's all bullshit, and you know it.	1	0	1
And how to delete the wrong or unwanted images uploaded.	0	0	0

978-1-6654-0816-5/22/$31.00 ©2022 IEEE

TABLE2: Sample Data

Id	Threat	Insult	Identity Hate
27aafc4eb33071b0	0	0	0
27d42ae876c0010a	0	0	0

Exploratory Data Analysis

Following data cleansing, the next step in any project is to get familiarized with data using EDA. Using a variety of featuresis one method to get this level of familiarization. Like by usingpie charts, bar graphs, correlation maps.

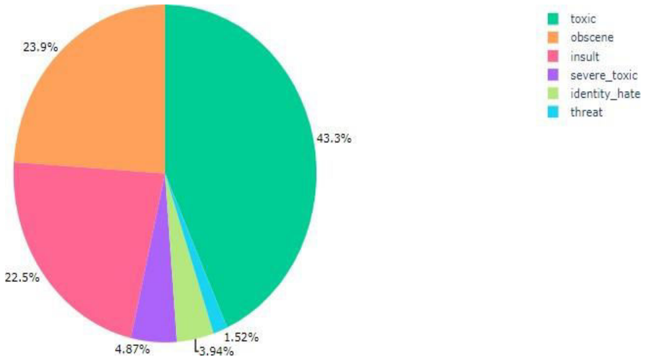


Figure 1 – Pie chart distribution from data collected

Here, each slice in the chart is concerning the size of that particular class in the data set. Figure1 infers that most comments in [5 6]data are toxic.

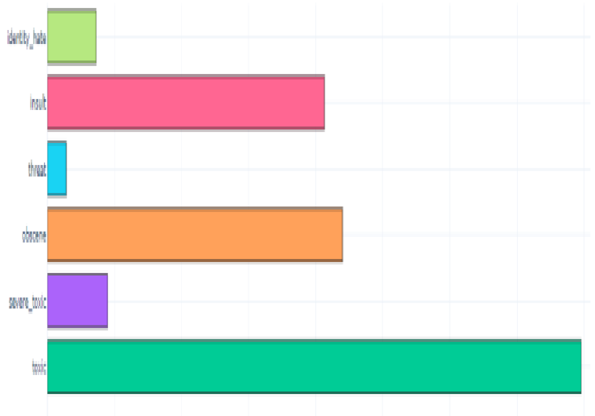


Figure 2 – Bar graph distribution from data collected.

IV. DATA PRE-PROCESSING

This is also known as Data Transformation. This includes data cleaning, dealing with null values, feature scaling, and so on. This is part of the data pipeline. Fig 2. As this project deals with text data, some more [4] pre-processing techniques aremandatory to apply to get efficient accuracy for the model. They are like dealing with punctuation removal, stop words removal, stemming, and so on.

Punctuation Removal:

Table 3 In this step, all punctuation from the text data is removed. There is an inbuilt library called a "string" in python which helps to remove punctuation from text data. It has some predefined characters such as @, (), [], +, -, /, *, # and so on.

TABLE3: Data after Punctuation Removal

Comment Text	Punctuation Removal
That's all Bullshit, and you know it.	Thats all Bullshit and you know it
And, how to delete the wrong or unwanted Imagesuploaded in 2018, June	And how to delete the wrong or unwanted Imagesuploaded in 2018 June

Lowering Text:

It's an easy and most adequate form of pre-processing in any text data. It's relevant to most NLP analyses. Table 4. It converts datato lowercase to have the same case preferably.

TABLE4: Data after Lowering Text

Comment Text	Lowering Text
Thats all Bullshit and youknow it	thats all Bullshit and youknow it
And how to delete the wrong or unwanted images uploaded in 2018 June	and how to delete the wrong or unwanted images uploaded in 2018 june

Tokenization:

Here, the text will be divided into minor units. Like sentences are converted into words. It divides a phrase or paragraphs into minor units, like words. Each minor word is called a token. Before pre-processing, it's necessary to classify the words that create a string. Tokenization is decisive because the essence of text could be efficiently understood here by inspecting words in a text.

TABLE5: Data after Tokenization

Comment Text	Tokenization
thats all bullshit and you,know, it	[thats, all, bullshit, and, you,know, it]

and how to delete the wrong or unwanted images uploaded in 2018 june	[and, how, to, delete, the, wrong, or, unwanted, images, uploaded, in, 2018, June]

Stop Word Removal:

Table 5. This helps us to remove unnecessary words from the text likeshe, them, he, for, it. The most commonly used words are removed here. These are eradicated from the text data beforetraining any models since they appear in plenty. This is doneby classifying text data into words and then removing wordsif they appear in stop words, which is an inbuilt function in NLP that contains a lot of unnecessary words. Table 6. This also helpsin focusing more on necessary data. NLP and ML to extract important information from data.

TABLE6: Data after Stop Word Removal

Comment Text	Stop Word Removal
thats all bullshit and you know it	bullshit know
and how to delete the wrong or unwanted images uploaded in 2018 june	delete wrong unwanted images uploaded june

Stemming:

It mostly expels the suffix from a word and diminishes it to itsroot word. This is a normalizing method in NLP for text data.It's a rule-based method since it divides the words from prefixes as per the necessity of words.

Spacy NLP:

It's an open-source library utilized in progressive NLP and ML to extract important information from data.

It's a method used in python to excerpt information from unregulated data to identify special features such as parts of speech (pos), Name entity recognition (ner), tokens, and so on. Spacy is a flagrant and easily implemented NLP library. It provides efficiency and dexterity and has a pro-active source. A relatively new technique for NLP was acquired by Matt Hannibal. It has many tasks utilized in NLP like parts of speech (pos), name entity recognition (ner), tokenization,

TABLE7: Spacy Features

Comment Text	Parts of Speech	Name Entity Recognition
thats all bullshitand you know it	[adverb, det, verb, CON, pronoun, verb, PREP]	[]
And how to delete the wrong or unwanted images uploadedin 2018 june	[CON, adverb, PREP, verb, det, adjective, CON, adjective,adjective]	[2018. june]

Table 7. Parts of speech (pos): It's like the development of semantics like a verb, noun, adjectives, adverb.

Name entity recognition (ner): It's like the dominant method of classifying name entities identified in the text to pre-defined clusters like the place, dates, persons. Spacy utilizes statistical techniques to classify the name entity recognition (ner).

Visualization with PyLDAVis:

It's a python library for collective topic model analysis. PyLDAVis is skeptical about how the algorithm was evaluated. To anticipate it equips topic-term distributions and elementary content about a corpus that the algorithm was evaluated on. Here, the predominant parameter is 'prepared' that will convert text data to other formats that are necessary for the analysis of PyLDAVis.

It's mainly developed to illustrate the different types of data that have been fit into the corpus of data. It's like the bilateral web-based analysis word2vec, and so on.

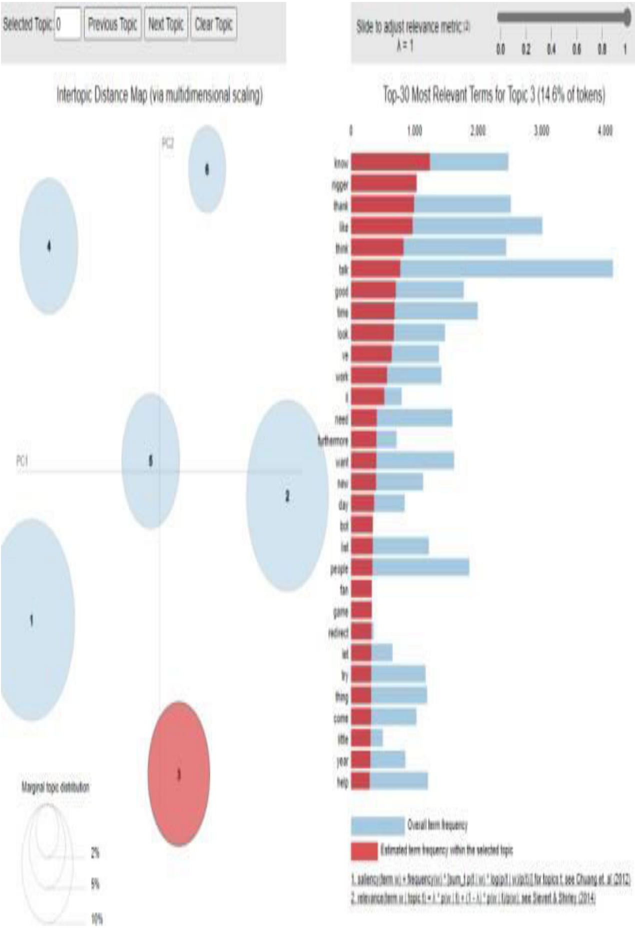

Selected Topic: 0
Previous Topic
Next Topic
Clear Topic
Slide to adjust relevance metric:(2)
λ = 1
0.0
0.2
0.4
0.6
0.8
1
Intertopic Distance Map (via multidimensional scaling)
Top-30 Most Relevant Terms for Topic 3 (14.6% of tokens)
PC2
PC1
6
4
5
2
1
3
Marginal topic distribution
2%
5%
10%
0
1,000
2,000
3,000
4,000
know
nigger
thank
like
think
talk
good
time
look
ve
work
i
need
furthermore
want
new
day
lol
let
people
fan
game
redirect
let
try
thing
come
little
year
help
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 3 – Overall Visualization of PyLDAVisDefault Topic Circles:

These are from the left side of the above Figure3, theyrepresent each topic in the data set.

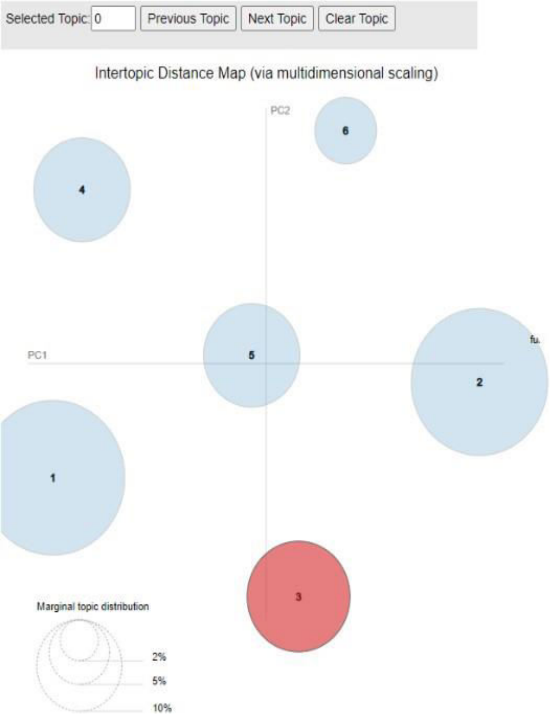

Selected Topic: 0
Previous Topic
Next Topic
Clear Topic
Intertopic Distance Map (via multidimensional scaling)
PC2
PC1
1
2
3
4
5
6
Marginal topic distribution
2%
5%
10%

Figure 4 – Visualization of Circles in PyLDAVis

Bar Graph:

When no topic is chosen, the bar graph basically will be in blue. When a particular class is selected it will be shown in redcolor. From this, the overall frequency of words in [8] a particularclass concerning overall data can be inferred, and also overall frequency alone can be inferred.

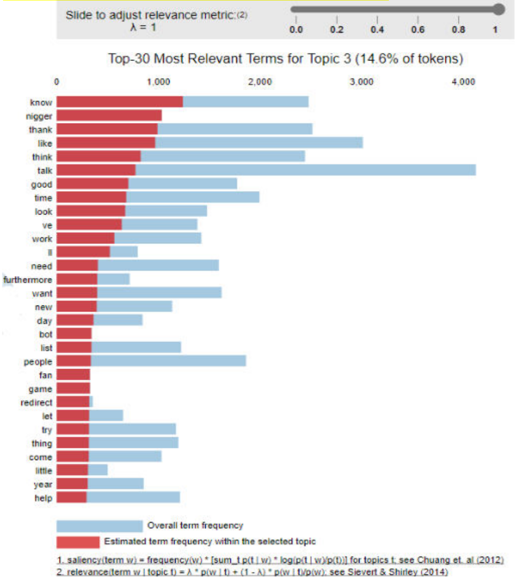


Figure 5 - Visualization of Bar Graph in PyLDAVis

V. METHODOLOGY

Interpretation of data was done in two phases. In Phase I, LSTM is applied data for training and to predict the test data parameters like type of class. And in Phase II, GRU is applied to data. Fig 4 and 5. After all analysis, this project is deployed using Heroku.

LSTM:

LSTM is called Long Short-term memory. LSTM's are easily implemented to avoid the long-term dependency complication. LSTM is combined of different states such as cell state and hidden state. It's like a distinct neural network. Data can be added or eradicated by using various gates [3]. A gate is analogous to layers that consist of various weights. Here, Algorithm analyzes the unnecessary information and it will be eradicated from the cell. This part of elimination was decided by the sigmoid function. It has mainly four [11]components. They are Input gate, Output gate, Forget gate, and cell state. On the last time step for the dense layer, LSTMdelivers a feature set, that may be utilized to generate results.

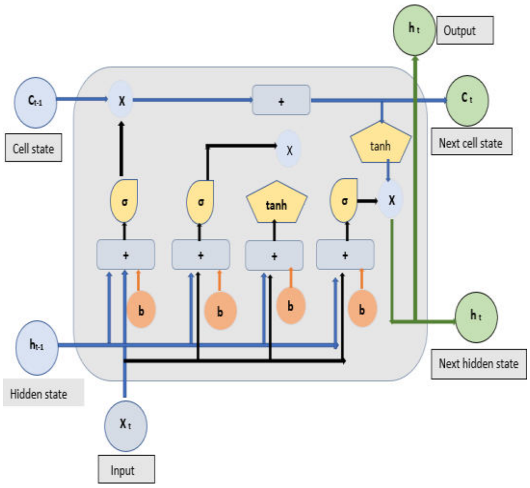


Figure6 – Structure of LSTM

$$f_t = \sigma\ (W_f[h_{t-1}, x_t] + b_f)$$

σ is the Sigmoid function, Weights matrices and bias are W_f and $b_{f.}$

f_t is the vector with values ranging from 0 to 1, concerning each number in cell state C_{t-1}. Sigmoid function determines if new data is necessary or not in fig 6., and tanh analyses weights to parameters that passed by, analyzing their level of necessity.These values are multiplied to renew the new cells. This is then joined to old memory C_{t-1} ensuring $C_{t.}$

$$i_t = \sigma\ (W_i\ [h_{t-1}, x_t] + b_i)$$

$$N_t = \tanh\ (W_n\ [h_{t-1}, x_t] + b_n)C_t = C_{t-1}\ f_t + N_t\ I_t$$

Here, C_{t-1} and C_t are cells concerning time periods t_1 and t. W and b are weights and biases.

$$O_t = \sigma\ (W_o\ [h_{t-1}, x_t] + b_o)h_t = O_t \tanh\ (C_t)$$

Here, W_o, b_o are weights and biases concerning output gates.

Output parameters h_t depend on output cell (O_t). Here, the sigmoid function determines which part is needed to make output and that output is multiplied by new parameters that areacquired by tanh layer from cell state (C_t).

TABLE8: Results From LSTM

	Precision	Recall	F1-Score	Support
0	0.81	1.00	0.95	39733
1	0.76	0.98	0.01	3385
Accuracy	----	----	0.94	36118
Macro avg	0.45	0.50	0.48	36118
Weightedavg	0.82	0.91	0.86	36118

From TABLE8, it is found that the LSTM model has obtainedan accuracy of 94%.

TABLE9: Sample Data of Predicted Test Data from LSTM

Id	Toxic	Severe toxic
00001cee341fdb12	1.0	0.0
0000247867823ef7	0.0	0.0

TABLE10: Sample Data of Predicted Test Data from LSTM

Obscene	Threat	Insult	Identity hate
1.0	0.0	1.0	0.0
0.0	0.0	0.0	0.0

From TABLE9 and TABLE10, these are the prediction resultsobtained for test data from LSTM.

GRU:

GRUs are enhanced versions of neural networks. In this analysis update and reset gates are utilized to solve the problem of vanishing gradient.

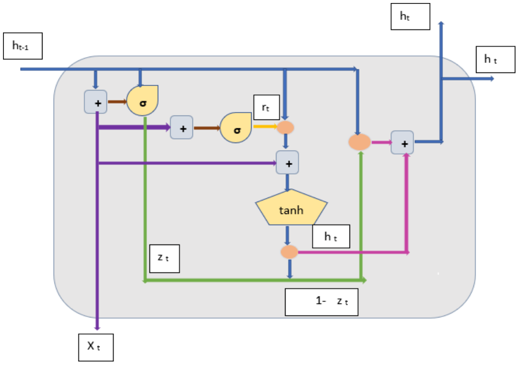


Figure7 – Structure of GRU

Here, Fig 7 the update gate controls the instructions that are progressing to memory, and the reset gate analyzes the data that flows out of memory [7]. These GRUs achieve well in sequence tasks. The architecture of GRU is simple as a neuralnetwork.

The weights are updated using back-propagation. Each gate isdetermined by using hidden state and bias from previous cellstates by reduction of variables. The activation function of GRU from previous [12] and current states (h_{t-1}) and h_t will operate at time-periods t-1 and t respectively.

$h_t = (1 - z_t) h_{t-1} + z_t h_t$

Here, z_t is an update gate that determines the necessity of a unit that amends the activation function.

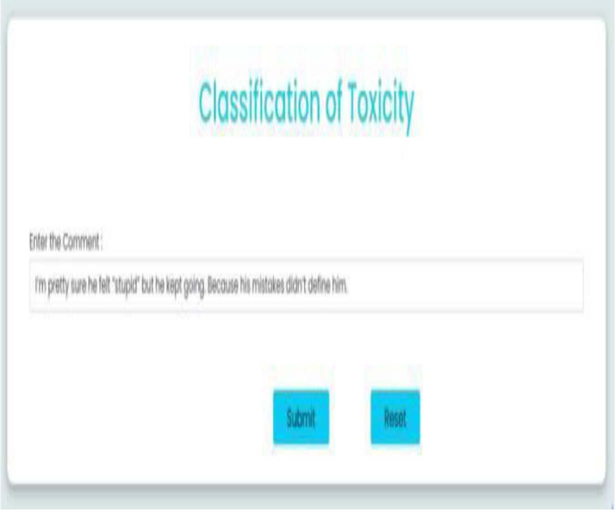
TABLE11: Results From GRU

	cision	all	Score	port
	5	0	2	6
	0	8	2	6
accuracy		-	2	2
Macroavg	3	3	2	2
weightedavg	3	3	2	2

From the TABLE11, it can be inferred that GRU model hasacquired an accuracy of 92%.

VI. DEPLOYMENT

It's a method of consolidating our models to an environmentwhere they can be deployed to a web app. Data is analyzed and evaluated using models like LSTM and GRU. Fig 8 & 9. Now, the model is ready for deployment. The testing application was done by using flask in vs code. Deployment was done by using Heroku and Git-hub.



Classification of Toxicity

Enter the Comment :

I'm pretty sure he felt "stupid" but he kept going. Because his mistakes didn't define him.

Figure8 – Web Page of our Project

Figure9 – Results from Web Page

VII. CONCLUSION

In this Paper, Analysis was done on toxic comments by usingNLP. This is done in two phases. In Phase I, the model is evaluated by using LSTM, and the class of toxicity is predicted for test data from LSTM trained model. The modelperformed well with an accuracy of 94%.

In Phase II, the model is evaluated using GRU and got an accuracy [13] of 92%. This project application is tested using a flask and deployed in the Heroku app that gives ultimate results.

VIII. FUTURE SCOPE

NLP is mostly like dealing with text data with various techniques. The future scope of this project is to develop a Classification of toxicity on images by enhancing and collecting image data from various social media platforms such as restaurant reviews, Facebook comments, and so on. Classifying the image data as toxic or non-toxic by utilizing Optical Character Recognition (OCR) is to be done in the future.

REFERENCES

[1] R. Vinaya Kumar, K.P. Soman and P. Poorna Chandran, "Long short-term memory based operating log anomaly detection," 2017 International Conference on Advances in Communicating, Communications and Informatics (ICACCI), 2017, pp. 236-242.

[2] R. Vinaya Kumar "Amrita-CEN-Senti-DB: Twitter Dataset for Sentimental Analysis and Application of Classical Machine Learning and Deep Learning.", TechRxiv (2020).

[3] Viswanathan S, Anand Kumar, Soman K.P, "A Sequence- Based Machine Comprehension Modelling Using LSTM and GRU." Emerging Research in Electronics, Computer Science and Technology. Lecture Notes in Electrical Engineering, vol 545, Springer, Singapore.

[4] P. Ram Manohar, "Toxicity of Ayurvedic Medicines and Safety Concerns: Ancient and Modern Perspectives, In History of Toxicology and Environment Health, Toxicology in Antiquity" (Second Addition), Academic Press, 2019, ISBN 9780128153390.

[5] Se, S., Vinaya Kumar, R., Kumar, M.A., Soman, K.P. "AMRITA – CEN@SAIL2015: Sentiment Analysis in Indian Languages. MIKE (2015).

[6] Kakuthota Rakshitha, Ramalingam HM, M Pavithra, Advi HD, Maithri Hedge, "Sentiment Analysis of Indian regional languages on social media," Global Transitions Proceedings, Volume 2, Issue 2, 2021, Pages 414-420.

[7] Guizhu Shen, Qingping Tan, Haoyu Zhang, Ping Zeng, Jianjun Xu, "Deep Learning with Gated Recurrent Unit Networks for Financial Sequence Predictions", 8th International Congress of Information and Communication Technology (ICICT – 2018).

[8] R. Dey and F.M Salem, "Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks," International Midwest Symposium on Circuits and Systems (MWSCAS), 2017, pp. 1597 – 1600.

[9] A. Akshith Sagar, J. Sai Kiran, "Toxic Comment Classification using Natural Language Processing," International Research Journal of Engineering and Technology (IRJET – 2020)

[10] Navoneel Chakrabarty, "A Machine Learning Approach to Comment Toxicity Classification," International Conference on Computational Intelligence in Pattern Recognition (CIPR 2019).

[11] P. Vidyullatha, Satya Narayanan Padhy, Javvaji Geetha Priya, Kakarlapudi Srija, Sri Satyanjani Koppisetti, "Identification and Classification of Toxic Comments Using Machine Learning Methods," International Journal of Research and Innovation in Applied Science (IJRIAS – 2021)

[12] Spiros V. Georgakopoulos, Sotiris K. Tasoulis Aristidis G. Vrahatis, Vassills P. Plagianakos, "Convolutional Neural Networks for toxic comment classification", Hellenic Conference (2018).

[13] Sara Zaheri, Jeff Leath, David Stroud, "Toxic Comment Classification," SMU Scholar, 2020