Multilingual Obnoxious Message Classification using Bidirectional Encoder Representation from Transformers (BERT)

Nikhil Banka
NSUT
New Delhi. India
bankanikhil29@gmail.com

Sparsh Narang
NSUT
New Delhi, India
sparshnarang@gmail.com

Mridul Gupta
Graphic Era Deemed to be University
Dehradun, India
mri.gupta@gmail.com

Abhay Sharma
Graphic Era Deemed to be University
Dehradun, India
abhayep03@ieee.org

Deepak Upadhyay
Graphic Era Hill University
Dehradun, India
deepaku.cse@gmail.com

2023 International Conference on Computer, Electronics & Electrical Engineering & their Applications (IC2E3) | 979-8-3503-3800-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/IC2E357697.2023.10262469

Abstract— **Everyday we witness incidents of social media bullying, harassment and various other toxic remarks. It has now become the part and parcel of social networking. Promotion of violence and hate speech, etc involve toxic comments in various forms on social media. The proposed model in this paper helps with the identification of all the obnoxious and questionable content on the internet by classifying the online comments as toxic or clean to tackle this problem. This is realised by using a pre-trained model - BERT. To enhance the performance, this model used RoBERTa (Robustly Optimized BERT Pre Training Approach), which has basically been formed by altering some hyperparameters which are tuned during the pre-training of BERT.**

Keywords—BERT, classification, RNN, LSTM, obnoxious

I. Introduction

In this era of Social Media we have seen many people are misusing it to outshow people. Not only this it is used to spread hate speeches against people, governments etc. In a recent case of "Bois Locker Room" [1], few schoolboys had used words which were not only obscene but a threat. Also, many people post lewd comments under the photos of Social Media Influencers. So, using this we aim at detecting different types of toxicity comments and classify them as threats, obscenity, insults, or identity-based hate.

Social networking platforms, being an inexpensive means of communication coupled with its capability to reach millions of users has now become an integral part in everyone's life (specially youngsters). What started off as an entertaining extra, has now become a social influence. The threat of harassment and verbal abuse in comments can hamper people from expressing themselves and in some cases, it can even take a toll on someone's mental health and put people at increased risk for suicide. This is why it becomes imperative to effectively facilitate non - toxic conversation in the comments of various public social media platforms.

Work done by Hochreiter, and J. Schmidhuber [3] published in 1997, is one of the earliest works on Long Short-Term Memory (LSTM), which helped us understand the working and application of mathematical formulae. In this paper, they have introduced LSTM design to overcome the error back- flow problems (vanishing gradient and exploding gradient). Implementation of LSTM by Sutskever et al. [4] where they demonstrated a simple implementation of the LSTM architecture to address various sequence-to-sequence issues.

Paper by Cho et al. [5] served as a reference for understanding recurrent neural networks (RNNs). Two networks were combined to form the RNN Encoder- Decoder. A sequence of symbols is encoded by one RNN into a vector representation having fixed-length, which is decoded into a different sequence of symbols by the other RNN. To maximise the conditional probability the source and target sequences were jointly trained.

Wang et al. [6] published a research article which laid emphasis on binary classification as a subset of sentiment classification. Regional CNN-LSTM model was proposed which consisted two parts viz., i) regional CNN ii) LSTM for predicting the VA (Valence and Arousal; dimensions to measure emotion) ratings of texts. This paper helped to understand the role of LSTM in dealing with language processing problems.

Vaswani et al. [7] gave an understanding of how language translation is implemented utilising a basic a transformer that purely worked on attention mechanisms and was far away from recurrence and convolution, to drastically reduce calculations. Work by J. Devlin et al. [8] was among the first papers to introduce Bidirectional Encoder Decoder Transformer (BERT). Here, the processing task was done on eleven different natural languages and BERT state-of-the-art brand-new outcomes were achieved. Although some literature talks of various filter designs [10-12] that can be applied for processing the image but the proposed BERT training mechanism works better.

Since BERT had shown great power on a variety of NLP tasks, Liu et al. [9] attempted to further enhance its performance by introducing RoBERTa (Robustly Optimized BERT Pretraining Approach). It proposed an improved strategy for pre-training BERT by modifying its hyperparameters. The model achieved state-of-the-art results and proved to be superior to BERT when training on large amounts of data for a long time and in terms of downstream task performance.

The presented work in this paper aims to use the knowledge of Natural Language Processing and BERT to identify if the input data is obnoxious or clean. Section-II gives the detailed methodology, section-III discusses results and finally the conclusion has been made in section-IV.

II. Methodology

A. Dataset

- *Training dataset*: The wiki corpus dataset has been used [2]. Wiki corpus is a collection of written texts, in this case the dataset specifically consists of sixty three million comments from across various public discussions on user pages and articles dating from the year 2004 - 2015. These comments were then classified into obnoxious or clean based on public ratings. The rating was accomplished by the process of crowdsourcing, wherein the dataset is rated on the basis of voting/poll by individuals.

- *Validation dataset*: Three different language comments viz., Italian, Spanish and Turkish from Wikipedia talk pages which are not in English language are considered.

- *Test dataset*: Here six different language comments viz., Turkish, Russian, Italian, Spanish, French and Portuguese from Wikipedia talk pages which are not in English language are considered. Make sure your template is the appropriate size for your paper first. This template has been designed to print on A4-sized paper. Please dismiss this file and download the Microsoft Word, Letter file if you plan to print on US letter-sized paper.

B. Exploratory data analysis (EDA)

The obnoxious content is not uniformly laid out across different categories, resulting in class imbalance issues. The training dataset comprises approximately 95 thousand comments with 21 thousand tags and around 86 thousand "clean" texts. This indicates that there are multiple categories pertaining to a single comment. For instance, it is possible for a comment to be classified as both "obnoxious" as well as "obscene".

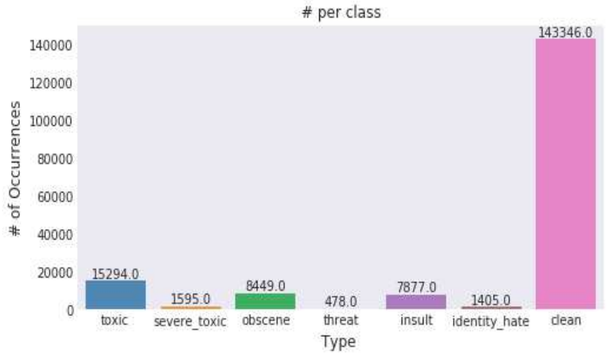


Fig. 1. *Class Wise Frequency Distribution of Comments in Training Dataset*

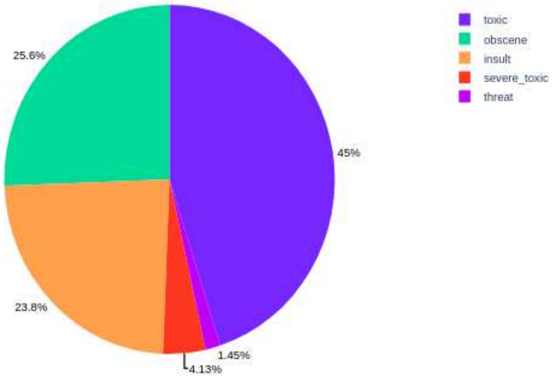


Fig. 2. *Pie Chart of Labels in Training Dataset*

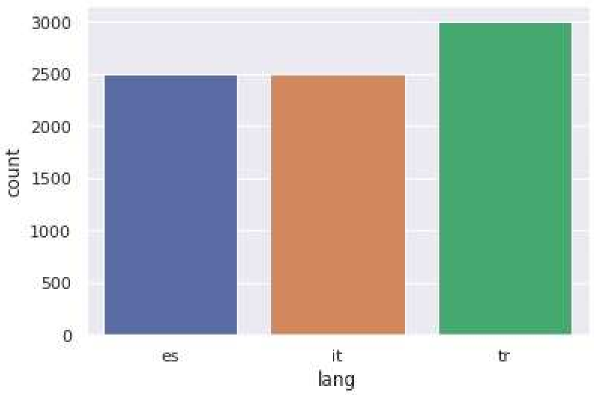


Fig. 3. *Language distribution of Validation dataset*

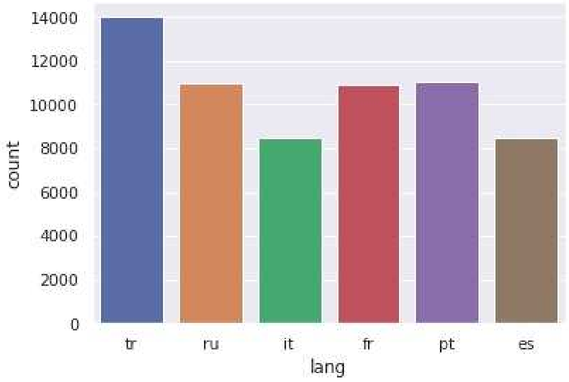


Fig. 4. *Language distribution of Testing dataset*

C. Word Cloud

Based on the type, since the comments are tagged in the following categories:-

i. Clean
ii. Toxic
iii. Severe_toxic
iv. Obscene
v. Threat
vi. Insult
vii. Identity_hate

Word Clouds for some categories have been attached for the reference (Fig. 5 to 11).

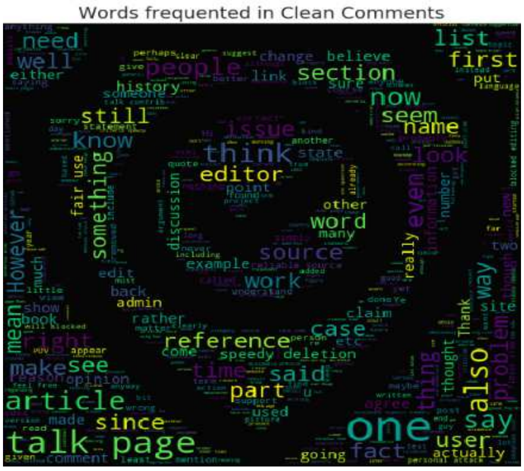


Fig. 5. *Words frequented in Clean Comment*

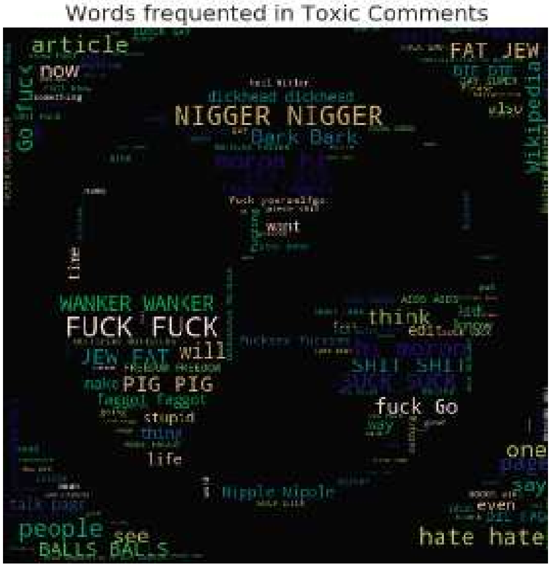


Fig. 6. *Words frequented in Toxic Comments*

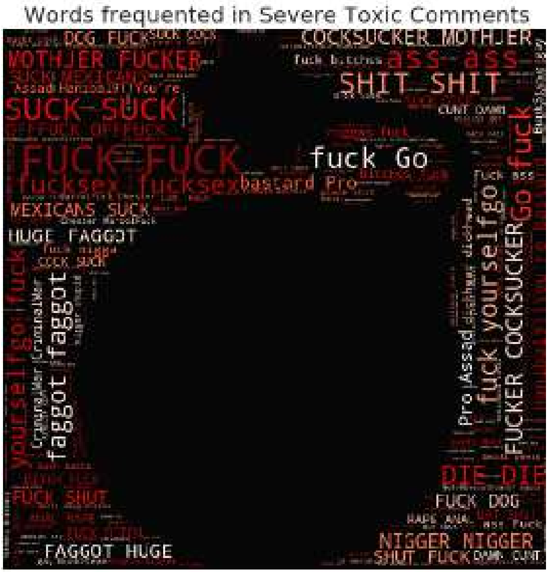


Fig. 7. *Words frequented in Severe Toxic Comments*

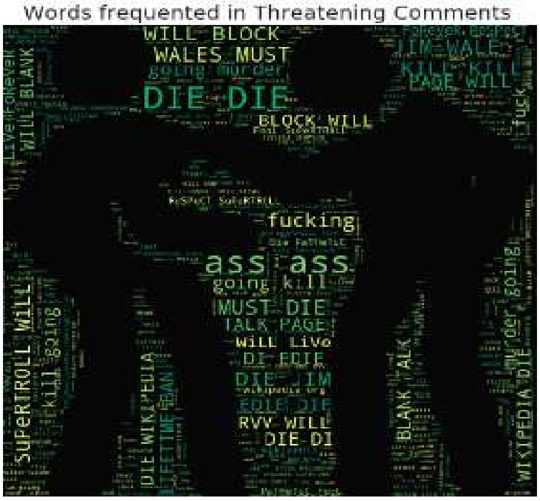


Fig. 8. *Words frequented in Threatening Comments*

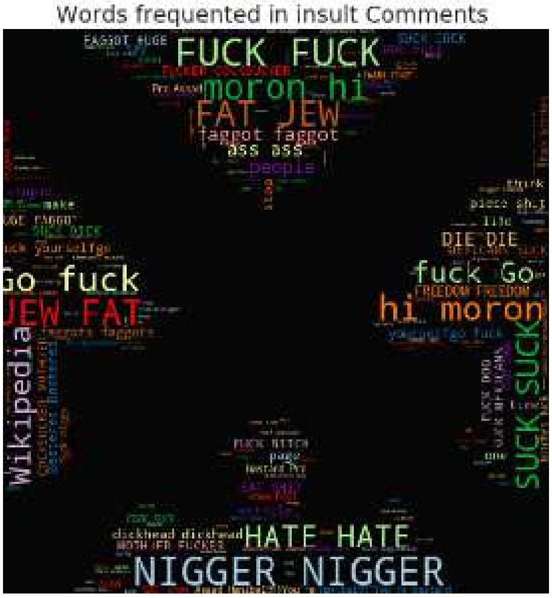


Fig. 9. *Words frequented in Insult Comments*

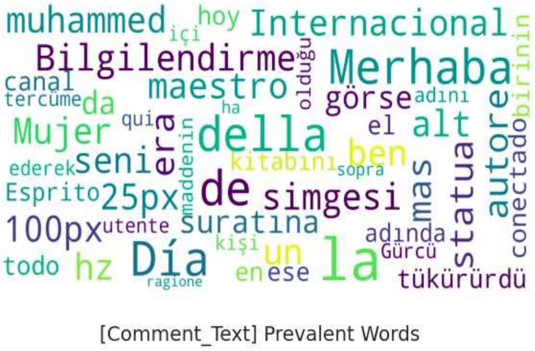


Fig. 10. *Words frequented in validation dataset*

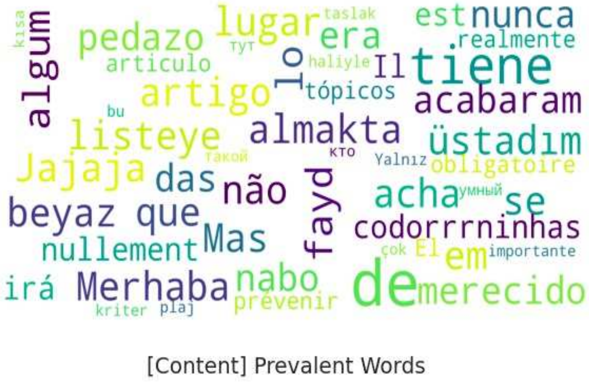


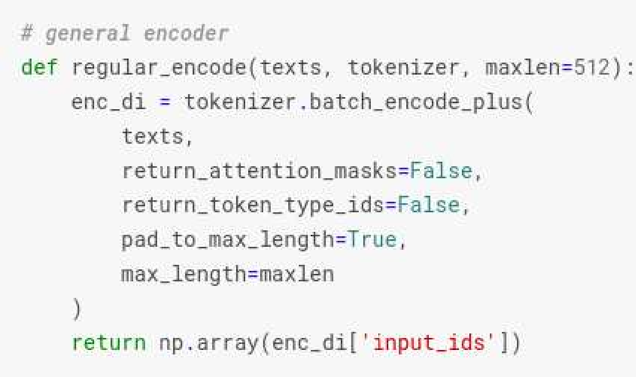
Fig. 11. *Words frequented in testing dataset*

The approach that we are taking is to feed the comments into the LSTM as part of the neural network, but we can't just feed the words as it is. So, this is what we did:

1. *Tokenization* - The first step is to break the sentence into words. Breaking down of a sentence into unique words is termed as tokenization. For example, the sentence "I love to read and write" becomes - ["I","love","to",”read”, "and","write"].
2. *Indexing* - The words derived from the above step are then given an index and put into a dictionary-like structure. For example, {1: “I”, 2: “love”, 3: “to”, 4: “read”, 5: “and”, 6: “write”}.
3. *Index Representation*- The sequence of words in comments are represented in terms of index and fed into the network. For example, [1,2,3,4,5,2].

D. Padding

We could make the shorter sentences as long as the others by adding extra zeros. At the same time, it is also possible to shorten the longer sequences and make their length similar to that of short ones. In such a case, the maximum length is set to 512.



```
# general encoder
def regular_encode(texts, tokenizer, maxlen=512):
    enc_di = tokenizer.batch_encode_plus(
        texts,
        return_attention_masks=False,
        return_token_type_ids=False,
        pad_to_max_length=True,
        max_length=maxlen
    )
    return np.array(enc_di['input_ids'])
```

E. Architecture of the model

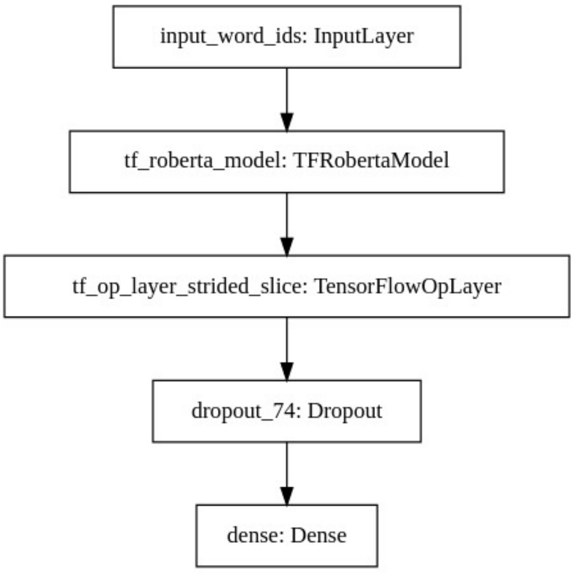

input_word_ids: InputLayer
tf_roberta_model: TFRobertaModel
tf_op_layer_strided_slice: TensorFlowOpLayer
dropout_74: Dropout
dense: Dense

Fig. 12. *Architecture of the model*

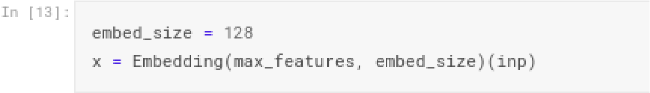
F. Embedding Layer

After the input layer, we pass the sequence of input to the Embedding layer. In this layer, we find the projections of the words on the vector space. The words are projected on the basis of distance of neighbouring words in a sentence. Embedding reduces the model size and prevents from dealing with very huge dimensions, in the case of using one-hot encoding to encode the words. This layer outputs the coordinates for each word in the vector space. For example, (-41.006) for "cat" and (-40.006) for "dog". The less the difference between the coordinates of two words, the more similar they are. This is how relevance and context can be detected.

Two strings can be input and then the distance between their projections can be calculated to detect if they are similar or not.

However, it poses a major drawback as it should also realise that “Apple” in “Apple is a healthy fruit” is a fruit and that in “Apple just released its new product” is a company and hence their projections should be different and away from each other.

The embedding size can be tuned as per the requirement of the experiment.



```
In [13]:
embed_size = 128
x = Embedding(max_features, embed_size)(inp)
```

G. Count based features(for unigrams)

There are three methods for producing count features in SKlearn for Python. Before producing a sparse matrix of word counts for every word in the phrase that is included in the dictionary, each of them first builds a vocabulary of words. Following is a brief description:

i. CountVectorizer
   - Generate a matrix showing the frequency of each word in the text.

ii. TF-IDF Vectorizer

- TF (Term Frequency) - The number of words (or terms) in the corpus of text (same as of Count Vect)

iii. IDF - Inverse Document Frequency -- penalises overused words. This can be regarded as regularisation.

iv. HashingVectorizer

- Replaces dictionary for vocabulary with a hashmap
- This makes it faster and more scalable for larger text corpus
- Provides multiple threads parallelism

In Fig. 13, TF-IDF is employed. The concatenated dataframe "merge" is used which includes text from both the train and test datasets to make sure that the vocabulary we develop includes all of the words that are specific to the testset.

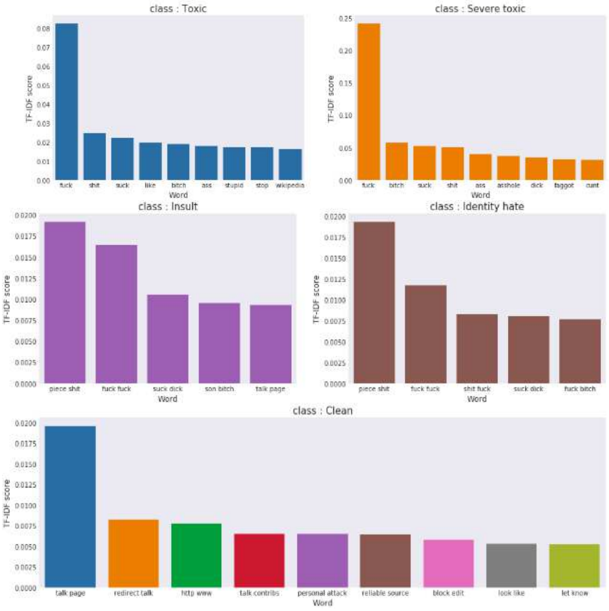
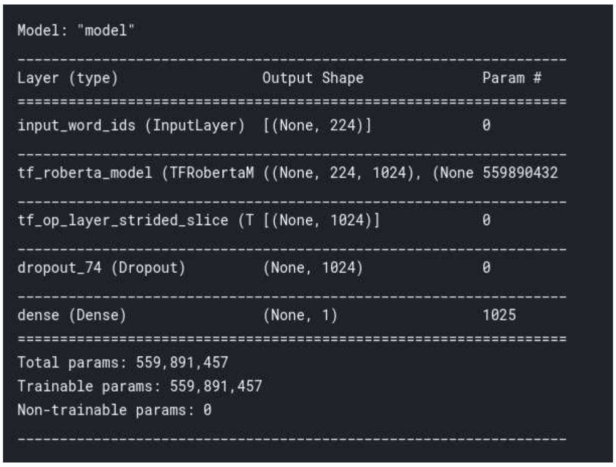


Fig. 13. *TF_IDF Top words per class(unigrams)*

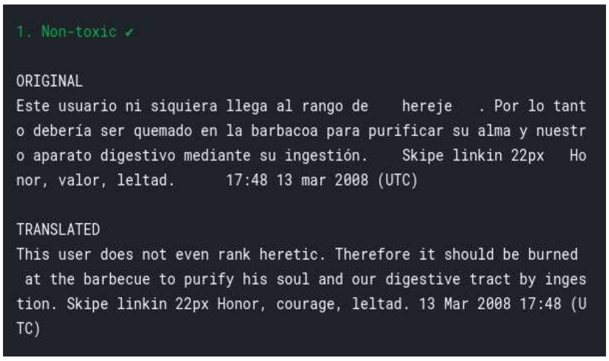
III. Results and Discussion

A. Model Summary



```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_word_ids (InputLayer)  [(None, 224)]             0
_________________________________________________________________
tf_roberta_model (TFRobertaM ((None, 224, 1024), (None 559890432
_________________________________________________________________
tf_op_layer_strided_slice (T [(None, 1024)]            0
_________________________________________________________________
dropout_74 (Dropout)         (None, 1024)              0
_________________________________________________________________
dense (Dense)                (None, 1)                 1025
=================================================================
Total params: 559,891,457
Trainable params: 559,891,457
Non-trainable params: 0
_________________________________________________________________
```

B. Visualizing the Model Performances



```
1. Non-toxic ✔

ORIGINAL
Este usuario ni siquiera llega al rango de    hereje   . Por lo tant
o debería ser quemado en la barbacoa para purificar su alma y nuestr
o aparato digestivo mediante su ingestión.    Skipe linkin 22px   Ho
nor, valor, leltad.      17:48 13 mar 2008 (UTC)

TRANSLATED
This user does not even rank heretic. Therefore it should be burned
 at the barbecue to purify his soul and our digestive tract by inges
tion. Skipe linkin 22px Honor, courage, leltad. 13 Mar 2008 17:48 (U
TC)
```

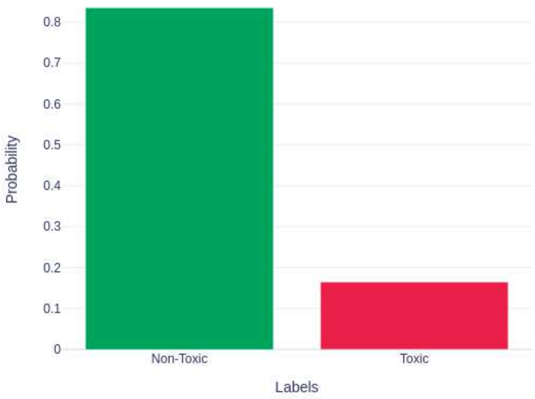
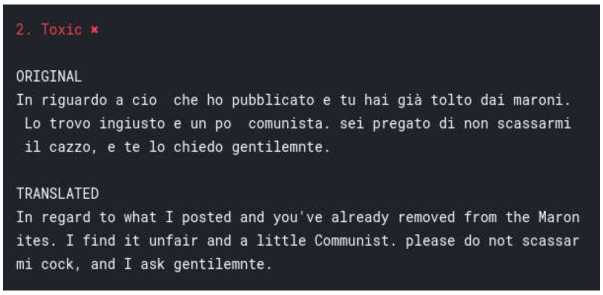


Fig. 14. *Prediction for validation comment #1*



```
2. Toxic ✖

ORIGINAL
In riguardo a cio  che ho pubblicato e tu hai già tolto dai maroni.
 Lo trovo ingiusto e un po  comunista. sei pregato di non scassarmi
 il cazzo, e te lo chiedo gentilemnte.

TRANSLATED
In regard to what I posted and you've already removed from the Maron
ites. I find it unfair and a little Communist. please do not scassar
mi cock, and I ask gentilemnte.
```

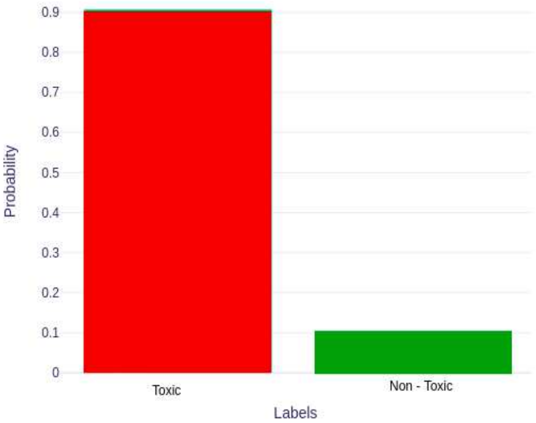
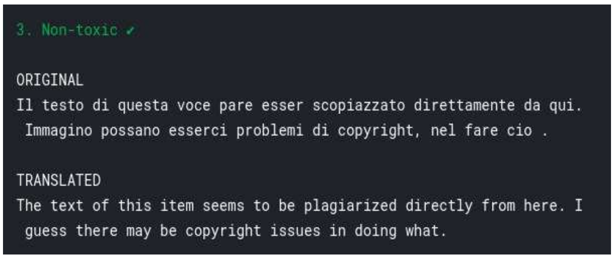


Fig. 15. *Prediction for validation comment #2*



```
3. Non-toxic ✓

ORIGINAL
Il testo di questa voce pare esser scopiazzato direttamente da qui.
 Immagino possano esserci problemi di copyright, nel fare cio .

TRANSLATED
The text of this item seems to be plagiarized directly from here. I
 guess there may be copyright issues in doing what.
```

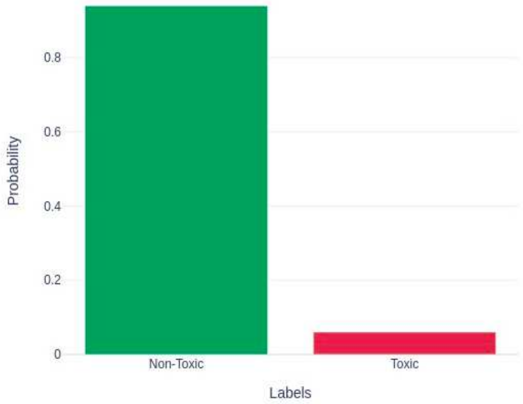


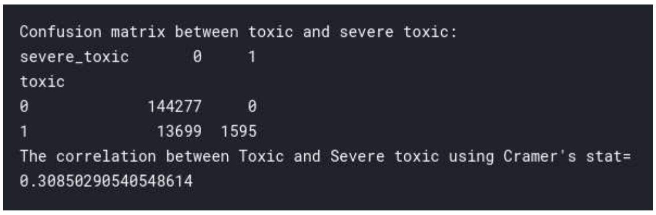
Fig. 16. *Prediction for validation comment #3*

C. *Accuracy*

Accuracy Achieved for the proposed model is found to be 93.06%.

D. *Confusion matrix*

The idea behind the confusion matrix is to describe the performance of the classifier used. This is achieved with the help of a set of test data whose true values are known apriori. In a nutshell, this table aids in predicting the performance of an algorithm.



```
Confusion matrix between toxic and severe toxic:
severe_toxic      0     1
toxic
0             144277     0
1              13699  1595
The correlation between Toxic and Severe toxic using Cramer's stat=
0.30850290540548614
```

IV. Conclusion

The proposed model outputs the category to which the input text belongs. First, the classification was performed on the dataset consisting of just English comments using Long Short-Term Memory (LSTM) and the accuracy was recorded as 98%. Further, improvisation was done on the proposed model to classify comments of multiple languages and accuracy of 93% was achieved.

References

[1] https://en.wikipedia.org/wiki/Bois_Locker_Room

[2] https://www.corpusdata.org/wikipedia.asp

[3] S. Hochreiter, and J. Schmidhuber, “Long short-term memory”, *Neural computation*, vol. 9, no. 8, pp.1735-1780, 1997.

[4] I. Sutskever, O. Vinyals and Q.V. Le, “Sequence to sequence learning with neural networks”, *Advances in neural information processing systems*, vol. 27, 2014.

[5] K. Cho et al. , “Learning phrase representations using RNN encoder-decoder for statistical machine translation”, *arXiv preprint arXiv:1406.1078*, 2014.

[6] J. Wang, L.C. Yu, K.R. Lai, and X. Zhang, “Dimensional sentiment analysis using a regional CNN-LSTM model”, *in Proceedings of the 54th annual meeting of the association for computational linguistics,* volume 2: Short papers, pp. 225-230, 2016, August.

[7] A. Vaswani et al., “Attention is all you need”, *Advances in neural information processing systems 30*, 2017.

[8] J. Devlin et al., “Bert: Pre-training of deep bidirectional transformers for language understanding”, *arXiv preprint arXiv:1810.04805*, 2018.

[9] Y. Liu et al., “Roberta: A robustly optimized bert pretraining approach”, *arXiv preprint arXiv:1907.11692*, 2019.

[10] Sharma A, Gupta M, Rawat T. FPGA Implementation of Lattice-Wave Half-Order Digital Integrator using Radix-$2^{r} $ Digit Recoding. In2022 6th International Conference on Electronics, Communication and Aerospace Technology 2022 Dec 1 (pp. 300-305). IEEE. DOI: 10.1109/ICECA55336.2022.10009223

[11] Sharma A, Gupta M, Rawat T. Optimal Design of Fractional-Order Digital Integrator using Lattice Wave Structure. In 2022 8th International Conference on Signal Processing and Communication (ICSC) 2022 Dec 1 (pp. 463-467). IEEE. DOI: 10.1109/ICSC56524.2022.10009297

[12] Gupta M, Sharma A, Upadhyay DK. Triple Notch Filter using Non-Uniform Transmission Lines for UWB Applications. In2022 6th International Conference On Computing, Communication, Control And Automation (ICCUBEA 2022 Aug 26 (pp. 1-5). IEEE. DOI: https://doi.org/10.1109/ICCUBEA54992.2022.10011019