IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AI BASED SENTIMENTAL ANALYSIS FOR SOCIAL MEDIA CONTENT

Mrs. Ashwini B
Assistant Professor, Computer Science and Engineering,
Knowledge Institute of Technology, Salem, India

Indhuja S[1], Kaviya S[2], Nisha C[3], Priyadharshini K[4]
Department of Computer Science and Engineering,
Knowledge Institute of Technology, Salem, India.

Abstract: Detecting toxicity on websites, considering both children and adults, entails recognising hazardous or distracting information. When it comes to children, the emphasis is on protecting them from explicit language, violence, and mature themes, which necessitates a comprehensive review of content to guarantee it satisfies age-appropriate standards. For adults, undesirable distractions may include abusive language, violent speech, or sexual content, which might detract from the intended user experience. Toxicology levels can be quantified and classified by constructing a toxicity detection system that uses web scraping, LSTM, and vectorization. This technology method allows users to make informed decisions about whether a webpage is appropriate for a certain audience, resulting in safer and more focused online environments. Through dynamic content analysis, the system provides insights.

Keywords - Toxic content classification, Long Short-term Memory (LSTM), Convolution Neural Network (CNN).

I. INTRODUCTION

The enormous use of online platforms and social media in today's modern world has resulted in undisputed connection, but it has also created a troubling issue: the predominance of abusive and toxic content in online debates. Maintaining a secure and polite online environment is critical, especially as the internet becomes an increasingly important aspect of everyday life. This project seeks to address this issue by creating an advanced system for detecting and classifying abusive information on websites.

The project takes an integrative approach, combining cutting-edge technologies such as Beautiful Soup for web scraping, Long Short-Term Memory (LSTM) networks for deep learning model training, and a vectorizer to efficiently convert words to arrays. Beautiful Soup allows for the extraction of textual content from webpages, establishing the framework for further research. LSTM networks, which are known for their capacity to grasp sequential data, are used to discover patterns and subtleties in language, improving the model's accuracy in identifying abusive and toxic language.The Flask web framework encapsulates the system's basic functionality, giving users with an intuitive interface. This integration allows users to enter webpage links, which prompts the system to scrape the content, evaluate its level of abuse and toxicity, and provide a clear label indicating whether the content is poisonous or non-toxic. The real-time classification mechanism provides prompt feedback, helping to create a safer online environment.

II. LITERATURE REVIEW

Influencer marketing has surged in importance within digital marketing strategies, with a notable 289% increase in dedicated influencer marketing platforms over four years, projecting a $6.5 billion industry. Influencers' ability to sway consumer behavior, commanding endorsement rates exceeding $200,000, has led to a rise in product placement approaches. Selecting suitable influencers is crucial for engagement and ROI optimization, aided by AI tools. The study examines how influencer type and product category alignment impacts engagement, emphasizing the effectiveness of both general and specialist influencers. Personalized product placement campaigns are vital for success, requiring collaboration between influencers and brands. While limitations exist in capturing real-time follower data, future research aims to address this using longitudinal data. Leveraging CNNs for image-based data classification showcases the study's commitment to advancing social media research.

Various adaptations of the BERT model, including SS-BERT, SO-BERT, and SSBERT+SOC, aim to enhance its capacity to understand nuanced language features related to subjectivity and identity. SS-BERT considers both subjectivity and identity terms, while SO-BERT focuses solely on subjectivity. SSBERT+SOC combines subjectivity and identity terms with a Sampling and Occlusion (SOC) regularization technique, potentially improving generalization and performance across different NLP tasks.To analyze bias, this study utilizes tools like the TextBlob3 library and SentiWordNet to generate subjectivity scores. However, manual inspection and validation of these tools' outputs on a sample of 80 data instances from four datasets were conducted.

The research suggests that employing a simple linear downstream structure is more effective than using more complex ones like CNN and BiLSTM. Fine-tuning a pre-trained language model with minimal hyperparameter adjustments improves the performance of toxic comment classification tasks, especially with limited datasets. Kowsari et al. proposed the Hierarchical Deep Learning approach for Text Classification (HDLTex), which utilizes hybrid deep learning architectures such as MLP, RNN, and CNN to achieve specialized comprehension at each document hierarchy level, enhancing text classification effectiveness.

Previous studies have explored the benefits of ensembles of classifiers, but this paper introduces an approach that combines various model architectures and different word embeddings specifically for toxic comment classification. Additionally, this work examines sentiment analysis in social media between 2014 and 2019, emphasizing its significance in interpreting customer opinions, understanding competitive market dynamics, and identifying trends in influencer marketing. The study highlights the transformative potential of AI-powered methods and neural networks in deriving practical insights from vast volumes of social media data.

III. EXISTING SOLUTION

Sentiment analysis in the context of social media analysis primarily uses machine learning or lexicon-based techniques, or both. Predefined dictionaries such as sent wordnet or TF-IDF are used in lexicon-based approaches, whereas machine learning techniques like SVM and Naïve Bayes need training data. However, study suggests that combining these techniques improves precision and effectiveness. Because the data on microblogging platforms especially Twitter is freely accessible to everyone in real time, it is the main source of data for sentiment analysis. This accessibility makes it possible for experts to gather a range of worldwide viewpoints.

On the other hand, In spite of having the greatest user base in the world, Facebook is not as popular for sentiment analysis because of its unstructured, unorganised data, which is full of misspellings and casual language use. Compared to Twitter, Facebook has different obstacles when it comes to content analysis, including sites, comments, and status updates. Because of these complications, sentiment analysis attempts face challenges, leading academics to favour platforms with more efficient, more structured data in order to conduct effective analysis. Other platforms, such as WordPress and YouTube, are consequently less preferred because of their restricted capacity for data extraction.

IV. PROPOSED SOLUTION

To tackle web page toxicity, we propose a comprehensive approach leveraging the strengths of LSTM and CNN models, with a focus on an ensemble model, LSTM-CNN, to achieve heightened accuracy. LSTM, known for its ability to capture long-term dependencies, and CNN, renowned for spatial feature extraction, offer complementary capabilities crucial for understanding the nuanced nature of toxic content. Our approach involves training LSTM and CNN models individually on a corpus of text comments, extracting their respective probabilities for toxic and non-toxic classes. Subsequently, the LSTM-CNN ensemble model combines these predictions, leveraging the collective wisdom of both architectures to make more informed decisions. By averaging the probabilities obtained from both LSTM and CNN models, the ensemble model produces a robust prediction, enhancing the overall classification performance. This fusion of LSTM and CNN enables the ensemble model to effectively discern toxic from non-toxic content, thereby contributing to a safer online environment. Through this proposed solution, we aim to provide a sophisticated and reliable method for web page toxicity classification, ensuring the protection of users from harmful online content.

Utilizing LSTM (Long Short-Term Memory) models for web page toxicity classification presents a promising approach. LSTM's capability to capture long-term dependencies in sequential data is paramount for understanding the context and nuances of toxic comments. By training LSTM models on a corpus of text comments, we extract their probabilities.

for toxic and non-toxic classes. This enables us to effectively discern between harmful and benign content. Leveraging LSTM's sequential processing, our solution offers robust classification, contributing to a safer online environment. Through the adoption of LSTM, we aim to provide a reliable and effective method for web page toxicity detection, prioritizing user safety.

V.**METHODOLOGY**

A. DATASET OVERVIEW

Within our evaluation framework, models are subjected to training and testing using binary class datasets. Unlike conventional methodologies where toxic comments are categorized into multiple subtypes such as hate speech, severe toxicity, obscenity, threats, insults, and non-toxicity, our approach simplifies the classification process into two distinct classes: toxic and non-toxic.

CATEGORY	NO OF WORDS	EXPERIMENTAL DATA
Non-Toxic	143812	114,850
Toxic	15759	12,607
Total	159571	127,457

Initially procured from a Kaggle repository, the dataset originally encompasses a multi-label structure, including designations like toxicity, severe toxicity, obscenity, threats, insults, and identity hate. Comprising a total of 159,571 comments, the dataset is divided into Non-Toxic (143,812 comments) and Toxic (15,759 comments) segments. Notably, a substantial class imbalance is observed, with toxic comments being underrepresented. To address this disparity, we conduct focused experimentation by randomly selecting 114,850 non-toxic comments.

B. DATA COLLECTION

Data collection forms a fundamental component of our methodology, facilitated by web scraping scripts leveraging the capabilities of Beautiful Soup. These scripts are meticulously crafted to navigate through web pages, parsing HTML content and extracting relevant text. The utilization of Beautiful Soup, a Python library renowned for its parsing functionality, ensures the efficient extraction of textual data from diverse web sources.

Web Scraping

Beautiful Soup, a Python library, serves as a powerful tool for web scraping, facilitating the extraction of data from HTML and XML documents with ease. Its user-friendly interface simplifies the process of parsing web pages, enabling developers to navigate through the document's structure and locate specific elements effortlessly. The web scraping workflow typically involves fetching the HTML content of the target web page, parsing it with Beautiful Soup to create a parse tree, navigating the tree to locate relevant elements, extracting the desired data, and finally processing and storing it for further analysis. This process is invaluable for tasks like data analysis, research, content aggregation, and more. However, it is essential to conduct web scraping responsibly, adhering to ethical guidelines and respecting website terms of service to avoid legal issues and maintain positive relationships with website owners.

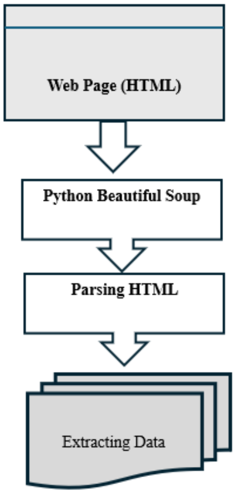


Fig 1. Architecture diagram for web scraping

C.DATA PREPROCESSING

Data preprocessing is an essential step aimed at enhancing the learning efficiency of machine learning models by cleaning and refining the dataset. The following steps are executed in a sequential manner to prepare the data for analysis:

1. Tokenization: This initial step involves segmenting the text into tokens, encompassing numbers, words, or symbols. Tokenization preserves crucial information necessary for accurate webpage classification.
2. Punctuation Removal: Stripping away punctuation marks, such as colons, question marks, commas, semicolons, and full stops/periods, optimizes the text for machine learning algorithms. This process is particularly beneficial for precise webpage categorization.
4. Stemming: Employing the Porter stemmer algorithm aids in simplifying word variations and transforming words into their root forms. This step minimizes redundancy and refines the data for accurate webpage classification.
5. Spelling Correction: Utilizing the 'pyspellchecker' Python library, misspelled words are corrected to add a layer of precision to the textual data. This is critical for maintaining accuracy in the context of webpage content analysis.
6. Stop Words Removal: Removing English stop words refines the text and streamlines machine learning models. This process decreases input feature complexity and contributes to more nuanced webpage classification.

These tailored preprocessing steps are integral to optimizing the dataset specifically for webpage analysis. They enable machine learning models to discern patterns effectively and accurately classify content in the dynamic context of web-based content categorization.

D.FEATURE ENGINEERING

Feature engineering optimizes machine learning models by crafting or selecting pertinent features from raw data. It involves techniques like creating new features, reducing dimensionality, selecting relevant features, encoding categorical variables, and extracting textual features. Dimensionality reduction methods like PCA streamline computation, while feature selection focuses on pertinent features. Textual feature extraction techniques like TF-IDF capture semantic information. Domain-specific knowledge enhances feature sets. Overall, feature engineering transforms raw data into meaningful representations, empowering models to make accurate predictions and derive insights.

E. PROPOSED METHODOLOGY

The process begins with gathering comments from diverse sources, meticulously organizing them in an Excel spreadsheet, and assigning labels denoting their toxicity status, distinguishing between toxic and non-toxic content. Subsequently, employing Natural Language Processing (NLP) techniques, the comment text undergoes rigorous cleaning and preprocessing to standardize and optimize it for analysis. This includes tokenization to segment text into meaningful units, punctuation removal, and stemming to reduce words to their root forms. With the text refined, relevant features are extracted to capture crucial information for classification. Techniques such as word embeddings transform words into numerical vectors, facilitating the representation of textual data in a format suitable for machine learning algorithms.

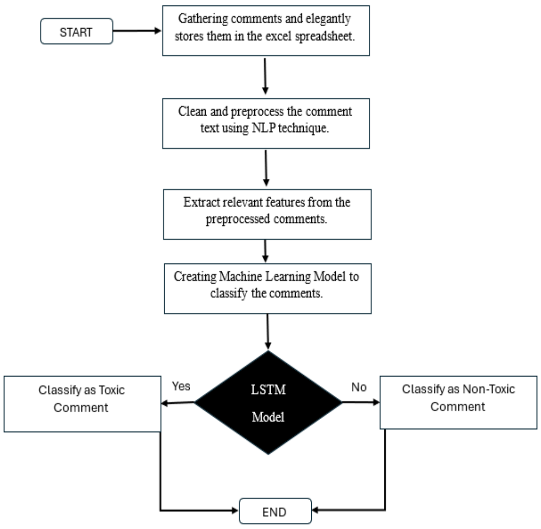

START
Gathering comments and elegantly stores them in the excel spreadsheet.
Clean and preprocess the comment text using NLP technique.
Extract relevant features from the preprocessed comments.
Creating Machine Learning Model to classify the comments.
LSTM Model
Yes
No
Classify as Toxic Comment
Classify as Non-Toxic Comment
END

Fig 2. Architecture diagram for proposed methodology

During the classification process, each preprocessed comment is fed into the trained LSTM model. The model predicts the probability of the comment being toxic, and if the probability surpasses a predefined threshold (e.g., 0.5), the comment is classified as toxic; otherwise, it is deemed non-toxic. This architecture seamlessly integrates data gathering, preprocessing, feature extraction, and model building stages to create a comprehensive solution for comment classification. Leveraging the LSTM's capabilities in understanding sequential patterns and long-term dependencies in text data, the model effectively discerns the underlying characteristics of toxic and non-toxic comments, ensuring accurate classification in various online platforms.

F. Long Short-Term Memory (LSTM)

LSTM networks were devised to address the inherent challenge of managing prolonged dependencies within recurrent neural networks (RNNs). In contrast to traditional feedforward neural architectures, LSTMs incorporate feedback connections, enabling them to process sequences of data comprehensively, such as time series, while retaining crucial information from prior points. This unique structural characteristic provides LSTMs with a notable edge in handling diverse sequential data types, including textual information, speech data, and time-series datasets. Their proficiency in capturing temporal dependencies makes them particularly effective for tasks necessitating a nuanced understanding of context over extended durations. Moreover, LSTMs demonstrate versatility in applications related to natural language processing, excelling in preserving context and capturing semantic associations within word or character sequences. They empower the model not only to evaluate individual terms but also to recognize syntactic relationships and parts of speech, offering indispensable contextual insights.

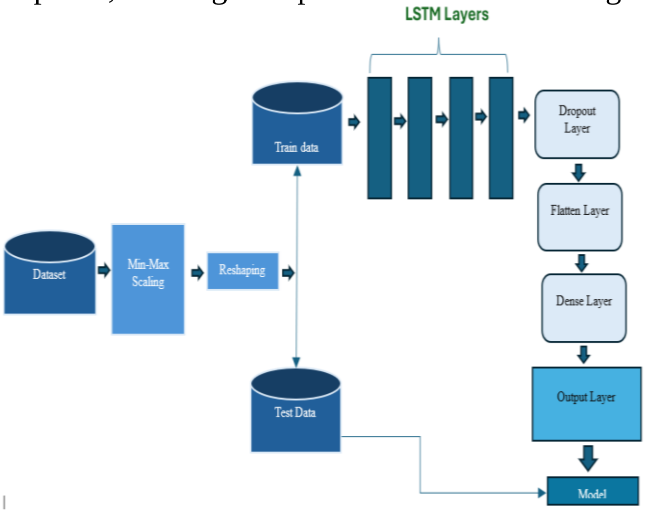


Fig 3. Architecture diagram for LSTM

Significantly, bidirectional LSTM layers assume a critical role in grasping the contextual nuances surrounding words. This contextual awareness proves pivotal in tasks like sentiment analysis, where discerning the intricacies of language is paramount for making accurate predictions. This is a result of people using social media in an informal manner. The dataset was heavily pre-processed to reduce its size and regulate it.

G. Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is a deep learning algorithm designed primarily for processing visual data such as images. It comprises multiple layers, including convolutional layers, pooling [illegible] it to input the

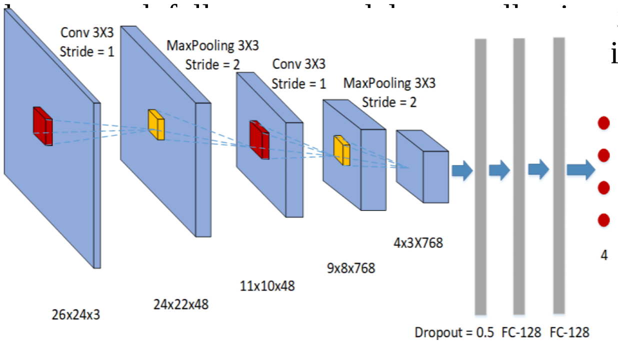


Fig 4. Architecture diagram for CNN

extracts feature like edges and textures, while pooling layers reduce spatial dimensions and enhance computational efficiency. CNNs leverage parameter sharing and local connectivity to capture spatial hierarchies within images efficiently. By iteratively learning from data through forward and backward propagation, CNNs can effectively classify images into various categories, making them widely used in tasks such as image recognition, object detection, and image segmentation.

H. LSTM-CNN

Combining Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) architectures, LSTM-CNN models offer a powerful approach for sequence-based data processing tasks such as text classification and sentiment analysis. LSTM layers excel in capturing long-term dependencies and sequential patterns, making them well-suited for analysing textual data. Meanwhile, CNN layers efficiently extract local features and patterns, enhancing the model's ability to understand spatial relationships within the input data. By integrating these two architectures, LSTM-CNN models can effectively process sequential data while also capturing spatial features, resulting in superior performance for tasks like sentiment analysis, where both sequential context and local features are crucial for accurate classification. Through a combination of LSTM's sequential learning and CNN's feature extraction capabilities, LSTM-CNN models provide a robust solution for a wide range of sequence-based tasks in natural language processing and beyond. The functionality of the proposed LSTM-CNN models, illustrating the amalgamation of LSTM and CNN for toxic comment classification. Considering LSTM and CNN as the two models and 'toxic' and 'non-toxic' as the two classes, predictions are determined using the following equation.

$$\textbf{Lstm-Cnn} = \textbf{argmax}\{\textbf{Toxic}_{\textbf{prob}}, \textbf{NonToxic}_{\textbf{prob}}\}$$

where argmax is used in machine learning for finding the class with the largest predicted probability. The Toxicprob and NonToxicprob indicate the joint probability of toxic and non- toxic classes by the LSTM and CNN models and are calculated as follows

$$\textbf{Toxic}_{\textbf{prob}} = \frac{\textbf{ToxicProb}_{\textbf{Lstm}} + \textbf{ToxicProb}_{\textbf{cnn}}}{\textbf{2}}$$

$$\textbf{NonToxic}_{\textbf{prob}} = \frac{\textbf{NonToxicProb}_{\textbf{Lstm}} + \textbf{NonToxicProb}_{\textbf{cnn}}}{\textbf{2}}$$

where ToxicProblstm, and ToxicProbcnn are the probability for toxic class by LSTM and CNN, respectively while NonToxicProbLstm, and NonToxicProbSVC are the probability scores for the non-toxic class by LSTM and CNN, respectively.

To illustrate the working of the proposed LSTM-CNN model, the values for one sample are taken from the dataset used for the experiments, given probabilities for the sample data are

- ToxicProbLstm = 0.7
- NonToxicProbLstm = 0.3
- ToxicProbCnn = 0.6
- NonToxicProbCnn = 0.4

The combined Toxicprob and NonToxicprob are calculated as

$$\text{ToxicProb} = \frac{0.7+0.6}{2} = 0.65$$

$$\text{NonToxicProb} = \frac{0.3+0.4}{2} = 0.35$$

EVALUATION METRICS

We assess the performance of machine learning models using key metrics, including accuracy, precision, recall, and F1 score.

1.ACCURACY

Accuracy represents the ratio of correct predictions to the total predictions made by the classifiers on the test data. The maximum accuracy score is 1, indicating all predictions from the classifier are correct, while the minimum accuracy score can be 0. The accuracy is calculated as

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

2.PRECISION:

The ratio of true positive predictions to the total number of positive predictions made by the model. It measures the accuracy of positive predictions.

$$\text{Precision} = \frac{T\,P}{TP+FP}$$

3. F1 SCORE

Precision and recall are not regarded as true representers of the performance of a classifier individually. F1 has been deemed more important as it combines both precision and recall and gives a score between 0 and 1. It is the harmonic mean of precision and recall and calculated using.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

IV RESULTS AND DISCUSSIONS

Numerous experiments are undertaken to assess the performance of selected machine learning classifiers and the proposed ensemble classifier. The experiments are categorized into three groups: those conducted without re- sampling, those with under-sampling, and those with over- sampling. The evaluation includes metrics such as accuracy, precision, recall, and F1 score to comprehensively analyze the classifiers' effectiveness. Additionally, the study investigates the impact of dataset re-sampling techniques on model performance, providing insights into the classifiers' robustness under different conditions.

Classifier	Accuracy
LSTM	96%
CNN	95%
LSTM-CNN	94%

Table 2. Performance results of all model on the dataset

A. PERFORMANCE ANALYSIS

The performance of the proposed LSTM-CNN model is compared against three state-of-the-art approaches, incorporating both machine and deep learning methodologies for toxic comments classification. Table-2 presents the performance evaluation results for LSTM-CNN alongside the other models. The findings demonstrate that the proposed LSTM-CNN outperforms other approaches in accurately classifying toxic and non-toxic comments.

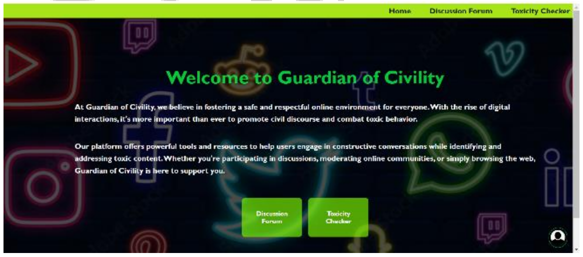
B. STATISTICAL T-TEST

To assess the significance of the proposed LSTM-CNN model, a statistical significance test, namely a T-test, is conducted. We establish two hypotheses to guide the analysis:

- Null Hypothesis: The proposed LSTM-CNN model exhibits statistical significance.
- Alternative Hypothesis: The proposed model does not demonstrate statistical significance.

The results of the statistical T-test indicate that the LSTM-CNN model is statistically significant across all resampling scenarios. Specifically, the null hypothesis is upheld for cases involving no resampling, under-sampling, and oversampling techniques.

C . SCREENSHOTS

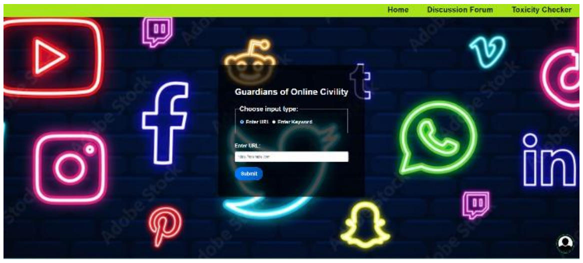


Welcome to Guardian of Civility

At Guardian of Civility, we believe in fostering a safe and respectful online environment for everyone. With the rise of digital interactions, it's more important than ever to promote civil discourse and combat toxic behavior.

Our platform offers powerful tools and resources to help users engage in constructive conversations while identifying and addressing toxic content. Whether you're participating in discussions, moderating online communities, or simply browsing the web, Guardian of Civility is here to support you.

Fig 5. Main Page



Guardians of Online Civility

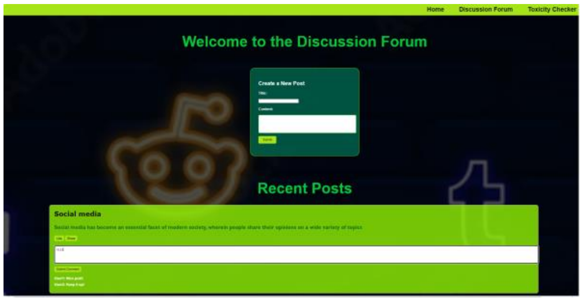
Choose input type:

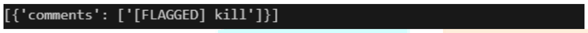
○ Enter URL ● Enter Keyword

Enter URL:

Fig 6. Toxicity Checker

Fig 7. Result of Toxicity Checker





```
[{'comments': ['[FLAGGED] kill']}]
```

Fig 8. Discussion forum where toxic comment will be flagged

VII. CONCLUSION

This paper assesses several machine learning models for classifying web-based toxicity and introduces an ensemble technique that combines LSTM for sequence modelling, Beautiful Soup for web scraping, and CNN for classification. The study systematically investigates the impact of imbalanced and balanced datasets, achieved through random under-sampling and over-sampling, on model performance through extensive experimentation. Two feature extraction methods, TF-IDF and Beautiful Soup, are utilized to generate feature vectors for model training. Results indicate suboptimal performance on imbalanced datasets, with balanced datasets improving classification accuracy. The proposed ensemble approach demonstrates robust performance, particularly with balanced and over-sampled datasets, mitigating the risk of overfitting associated with imbalanced datasets. TF-IDF outperforms in classifying toxic comments. The ensemble approach exhibits efficiency for both toxic and non-toxic comments, achieving superior performance with an accuracy of 0.97 when combined with TF-IDF features. Comparative analysis with cutting-edge approaches validates the enhanced performance of the ensemble, demonstrating its adaptability to feature vectors of varying sizes. While offering improved efficiency, the proposed ensemble method presents higher computational complexity, warranting further investigation in future research. Moreover, the influence of dataset imbalance on reported accuracy is acknowledged, emphasizing the need for caution. Future research endeavours include experimentation with multi-domain datasets and an expanded dataset collection for comprehensive toxic comment classification evaluations.

VIII. REFERENCES

[1] M. Umer, I. Ashraf, A. Mehmood, S. Kumari, S. Ullah, and G. S. Choi, (2021) __Sentiment analysis of tweets using a unified convolutional neural network-long short-term memory network model,'' Comput. Intell., vol. 37, no. 1, pp. 409–434.

[2] Zhixue Zhao, Ziqi Zhang, and Frank Hopfgartner (2021): "A Comparative Study of Using Pre-trained Language Models for Toxic Comment Classification" published in the Companion Proceedings of the Web Conference 2021.

[3] Matthew N. O. Sadiku., Tolulope J. Ashaolu., Abayomi Ajayi-Majebi., Sarhan M. Musa "Artificial Intelligence in Social Media". Journal of January 2021 International Journal of Scientific Advances 2(1).

[4] F. Rustam, M. Khalid, W. Aslam, V. Rupapara, A. Mehmood, and G. S. Choi,(2021) A performance comparison of supervised machine learning models for COVID-19 tweets sentiment analysis,'' PLoS ONE, vol. 16, no. 2, Art. no. e0245909.

[5] M. S. Basha, S. K. Mouleeswaran, and K. R. Prasad, (2021) Sampling-based visual assessment computing techniques for an efficient social data clustering, '' J. Supercomput., pp. 1–25.

[6] R. Beniwal and A. MMaurya, (021) "Toxic comment classification using hybrid deep learning model" in Sustainable Communication Networks and Application. Cham, Switzerland: Springer, pp. 461–473.

[7] Hong Fan 1, Wu Du., Abdelghani Dahou., Ahmed A. Ewees ., Dalia Yousri ., Mohamed Abd Elaziz., Ammar H. Elsheikh ., Laith Abualigah ., & Mohammed A. A. Al-qaness . "Social Media Toxicity Classification Using Deep Learning:Real-World Application UK Brexit" Journal of Electronics 2021, 10(11), 1332.

[8] B. Pan, Y. Yang, Z. Zhao, Y. Zhuang, D. Cai, and X. He (2019): "Discourse Marker Augmented Network with Reinforcement Learning for Natural Language Inference" available on arXiv as a preprint (arXiv:1907.09692).

[9] S. R. Basha and J. K. Rani, (2019) “A comparative approach of dimensionality reduction techniques in text classification, “Eng., Technol. Appl. Sci. Res., vol. 9, no. 6, pp. 4974–4979.

[10] Spiros V. Georgakopoulos, Sotiris K. Tasoulis, Aristidis G. Vrahatis, and Vassilis P. Plagianakos (2018): "Convolutional Neural Networks for Toxic Comment Classification" presented at the 10th Hellenic Conference on Artificial Intelligence, pages 1-6.

IJCRT