Naif Arab University for Security Sciences
Journal of Information Security and Cybercrimes Research
مجلة بحوث أمن المعلومات والجرائم السيبرانية
https://journals.nauss.edu.sa/index.php/JISCR

JISCR

Safeguarding Online Communications using DistilRoBERTa for Detection of Terrorism and Offensive Chats

Mohamed Safwan Saalik Shah[1,*], Amr Mohamed Abuaieta[2], Shaima Saeed Almazrouei[2]
[1]Middlesex University Dubai, Dubai, UAE.
[2]Digital Forensics Expert at International Center for Forensic Science and Criminology, Dubai Police, Dubai, UAE.

Received 25 Feb. 2024; Accepted 22 May. 2024; Available Online 29 Jun. 2024

Abstract

People use social media for both good and distasteful purposes. When used with malicious intent, it raises significant concerns as it involves the use of offensive language and hate speech that promote terrorism and other negative behaviors. To create a safe, secure and pleasant environment, these communications must be closely monitored to prevent severe problems, associated risks and other pertinent issues. With the help of AI, specifically Large Language Models (LLM), we can quickly analyze text and speech to determine whether the communications promote the dangers identified here above not to mention other toxic elements. For this research, the LLM used is the DistilRoBERTa model from the Transformers library using Hugging Face. The DistilRoBERTa model was trained on datasets consisting of terrorism-related conversations, offensive-related conversations, and neutral conversations. These datasets were obtained from publicly available sources. The results of the experimentation show that the model achieved 99% accuracy, precision, recall, F1 score, and ROC curve. To improve the robustness of the model, it must be continuously fine-tuned to predict dynamic communication behavior since real conversations are inaccessible due to restrictions. A drag-and-drop interface is used to upload the files and get the categorical output, ensuring seamless and easy interaction.

I. Introduction

Social media is used by individuals for online communication between friends, families, colleagues, and the world for various purposes such as chatting, sharing information, conveying thoughts, and business applications. There are around 3.78 billion users on different social media platforms [1]. Due to this large user base, there are concerns about the ways in which social media is used. Many use it for business, education, leisure, politics, and other domains. In contrast, others use it with harmful intentions such as spreading hatred, causing corruption, engaging in illegal activities, promoting cyberbullying, and condemning different groups based on race, religion, or other factors. This can be categorized as terrorism, offensive language, hate speech, and other types. Terrorism texts use attack terms, violence, threats, plots, recruitment, and propaganda. Terrorism words include but are not limited to words such as shoot,

Keywords: Social media, Offensive language, Terrorism, Large language models, DistilRoBERTa model.

Production and hosting by NAUSS

* Corresponding Author: Mohamed Safwan Saalik Shah
Email: saalik.shah011@gmail.com
doi: 10.26735/VNVR2791

bomb, assassinate, guns, explosion, hijack, and others. Offensive language are texts that use profanity. Offensive words include but are not limited to words such as bullshit, faggot, bastard, swine, and others. Hate speech are texts that use discriminatory words and spread hate-related speech. As we all know, terrorism is widespread in different parts of the world. It causes unnecessary and unwanted corruption and havoc to society, which needs to be controlled and prevented. We also find that many people using social media and other platforms tend to use hate speech, creating unwanted arguments and problems that result in undesired outcomes [4],[7].

Technology and communication play a crucial part in the promotion of terrorism as they are used to inform attack plans, recruit individuals, advocate hatred through speeches, misguide, and spread information to different individuals of different age groups to join terrorism-related activities. Offensive speech is used to harass and bully individuals based on their race, religion, gender, or other characteristics. For a long time, governments and other delegated authorities such as social media providers, through their limited access to legally tracking social media communications, especially messaging apps for conversations between individuals and groups, to monitor their behavior on whether they exhibit negative conduct cannot perform advanced administrative methods due to the strict privacy regulations adhered to which limits the authorities to surveil effectively. Depending on the seriousness, these regulations can be exempted. But another issue faced is monitoring or inspecting the communication manually. This approach is a tedious process due to numerous conversations and communications that take place between many individuals and groups, making it difficult to manually perform these tasks to identify whether the communication is categorized as terrorism, offensive, or neither [10],[11].

To resolve this issue, we can implement AI, specifically Large Language Models (LLM), to help identify and categorize whether a particular conversation is related to cyberbullying, terrorism, offensive language, racism, or neither [3]. LLMs are deep learning models trained through a corpus of online data. Through this rigorous training, the models can understand the semantic context of sentences and words, produce meaningful content, explain contents to users, and classify data based on given categories. Leveraging these features, LLMs can be used for specific use cases containing speech or text and configured to produce specific outputs. With the help of LLMs, we can train the dataset on texts, tweets, and conversations to categorize content as illegal activities, racism, terrorism, offensive language, or neither. We can then test the model to analyze different conversations from various sources and identify whether these communications encourage terrorist activities, spread offensive language, or promote neither. This can help prevent terrorist attacks and activities at an early stage and prevent misuse of social media that condemns individuals based on their physical or personal differences [3],[6].

A major issue faced in implementing LLMs is access to datasets that help distinguish between the types of conversations such as cyberbullying, terrorism-related, offensive language, and neither. These are restricted information and are not released for public use since there is a possibility that terrorists and other individuals with malicious intentions may use it to understand what keywords or indicators are used to categorize a particular input. The background of the problem is that inspecting communication on social media platforms would take a massive amount of time and labor. It is a time-consuming process that requires human intervention to identify and categorize the type of conversation due to the large volume of conversations between individuals.

The purpose of the study is to find an automated approach that can categorize whether a file is related to terrorism, offensive language, or neither by using an approach that would be an equal substitute for human intervention to investigate which category the text file belongs to, thus decreasing the time and labor required.

The aim of the research is to create an automated approach using LLMs, specifically

the DistilRoBERTa model, to analyze text files containing chats found from various sources and categorize them as terrorism, offensive language, or neither. To create easy human interaction with the LLM, a drag-and-drop interface will be used to upload the text files and get the category of the text file as output for each of the files uploaded.

The DistilRoBERTa model was chosen based on three important aspects: open source, efficiency, and compatibility. DistilRoBERTa is open-sourced, freely accessible, and requires no usage costs unlike GPT-4. This makes it a viable option for research and experimentation. DistilRoBERTa, being a distilled model, is efficient since it provides faster inference which helps classify inputs faster and requires low computational footprint for training unlike other models such as GPT-4 and Mistral-7b which require high-end GPUs for training. DistilRoBERTa can easily be fine-tuned on specific datasets for performing tasks without the requirement of high computation resources. By choosing DistilRoBERTa, we are using a model that is freely accessible without any usage costs and is resource efficient, requiring low memory usage. This makes it an ideal choice for this experimentation without having any significant overhead unlike GPT-4, LLaMA, Mistral-7b, and BERT models.

The objectives of the research are as follows:

1) Identify datasets that will be useful for training since terrorism and offensive-related chats are not available in public sources. Synthetic data will also be created and used to retrain the model after the training and testing phase to improve the robustness of the model against unprecedented data.
2) The datasets should undergo data manipulation and data preprocessing to produce quality data, and the dataset should be balanced to avoid bias within the dataset.
3) The DistilRoBERTa model will be trained on the training samples, validated on validation samples, and tested on the testing samples to evaluate the model's accuracy.
4) The evaluated model will be implemented using Python Flask by creating a drag-and-drop interface for uploading files that will classify and categorize them based on the file type.

The significance of the research is that if the LLM model can achieve excellent results, it can be used for PDFs, OCR images, and other types to detect whether they contain terrorism or offensive-related data. The scope of the project is restricted to publicly available datasets that are mainly extracted from Twitter and curated from other sources that can be categorized as offensive and terrorism-related texts. Synthetic data will also be created to fine-tune the model based on incorrect predictions of the model and will be a continuous process. Access to actual conversations is a challenge since this type of data is not published publicly since individuals with bad intentions can use it for the same reasons and create models to identify whether their conversations would be classified as terrorism and if it does they will converse using different semantic words to avoid detection.

The research paper will go as follows: first, we will look at the research done in the field of Machine and Deep learning to detect terrorism and offensive texts. We will then identify the approach to be taken based on the pros and cons of the research findings and focus on the areas where there is room for research. The next step is to define the approach to be taken and the methods that will be used. Then, the next step is to describe the implementation resources and the procedure. Finally, we will look at the results and compare them to what was found in the literature review, as well as go over the limitations and conclusion.

II. Literature Review

Rajendran et al. [1] and Nithyashree et al. [13] conducted research on extremism and terrorism on social platforms by using LLMs. Their experimentation found that the BERT model performed quite well when identifying one category, which was toxic content classification, achieving 98% accuracy, while the RoBERTa model performed the best overall, achieving 95% accuracy.

Among the researchers who used the Machine Learning model called SVM (Support Vector Machines) to identify toxic texts in social media, Abijith and Prithvi [4] were able to achieve 94% accuracy, while Čepulionytė et al. [6] used a different approach and created four categories (aggressive, insulting, toxic, and malicious), achieving 85-87% accuracy. Gaikwad et al. [12] focused on looking at areas where other researchers have not explored, such as different terrorism ideologies, and implemented methods to get as much information as possible, such as the use of the n-grams technique. The experimentation resulted in 70% accuracy. Alshalan and Al-Khalifa [15] focused their research on hate speech and harassment and were able to achieve 97% accuracy using the SVM model.

Another group of researchers experimented with the Machine Learning model called KNN (K-Nearest Neighbors). Mussiraliyeva et al. [7] proposed a solution to detect religious extremism messages in the Kazakh language and ensured that the dataset used had an equal number of labels. The experimentation resulted in 99.6% accuracy. Pais et al. [8] conducted research on determining the best ML algorithm for detecting extremism and radicalization using user sentiment analysis and text mining. Their research showed that KNN was the best model since it helped effectively reduce the noise present in the dataset. Shirsath et al. [11] conducted research on children's use of social media by using a system to monitor their activities and contents. By using KNN, they planned to recommend content that is harmless. They were able to achieve 89% accuracy.

Some researchers performed a different approach. Hussain and Mohideen [2] conducted research to determine several crime categories such as kidnapping, murder, bribery, and others in multilingual languages. The research involved using 49 languages, and AI was implemented using an encoder and decoder to translate all the languages into English. The research was done using a Machine Learning model called the Naïve Bayes model, where the model got a precision score of 95% and a recall score of 98%. Sharif et al. [3] proposed a solution using Machine Learning, specifically SGD (Stochastic Gradient Descent) with TF-IDF (Term Frequency – Inverse Document Frequency) using unigram and bigram techniques to classify toxic/offensive chats in Bengali text. The model achieved 84% accuracy. Fkih et al. [9] conducted research on the use of offensive speech in the Arabic language. The dataset used in this research was balanced using the SMOTE technique. The experimentation showed that the random forest model achieved the highest accuracy score of 90%. Shevtsov et al. [14] conducted research on the spread of false information and manipulation by bots using fake accounts on Twitter. To combat this issue, the authors introduced a Semi-Automatic Machine Learning Pipeline (SAMLP) to detect Twitter bots. The results of the experiment show that the model performed 10% better than existing bot detectors, with the F1 score averaging 83%. Gaikwad et al. [5] conducted a research survey where they performed a systematic review of different research papers related to online extremism. The research examined 64 studies that used 64 different datasets and different ML algorithms. The study found that in three papers, AdaBoost achieved 99% accuracy, which was the highest. Fahim and Gokhale [10] focused on the research of negative impacts caused by extremism and radicalization. They experimented using artificial neural networks and achieved 90% accuracy.

Based on the literature review, many authors have explored languages other than English. Many have implemented ML models instead of LLM models, likely due to the research findings they examined and the fact that LLM is a relatively new concept. Regardless of the models used, some performed well with accuracy scores over 99%. The downside of the research includes imbalanced datasets that limit the model's scope, not using LLM in their research, and the limited test use cases contributing to less reliable models. However, they could perform better if more complex data with better context is used, whether created or obtained. In terms of the categorizations, the authors used different approaches, such as classifying in

terms of the level of intensity (low, moderate, high) in hate speech, terrorism, kidnap threat, bullying, and other types. For this research, we will focus on three types: terrorism, offensive, and Neither, as illustrated in Fig. 1 Data Categorization.

After analyzing the pros and cons of the research explored on detecting terrorism and offensive chats using AI, as shown in Fig. 2 Proposed Solution. The data pipeline and model that is to be created needs to address the cons of the data pipeline and models found in the research, which are:

1) Handling Imbalanced Dataset: This is done by implementing data manipulation methods to over-sample or under-sample the respective classes, thus balancing the data.
2) Avoiding Bias Datasets: This is achieved by using different datasets extracted and curated from different sources to cover different ideologies and perspectives.
3) Implementing State-of-the-Art AI Models: The current AI model that tends to perform well with text and speech is LLM since it has proved helpful due to its remarkable training on the corpus of online data.
4) Robustness of the Model: The model may still have issues determining the context and may need clarification. The misclassified data should be examined to create synthetic data to train the model to understand the contents better and improve its interpretation.

Based on this assessment, the LLM model will be created accordingly by addressing all the issues mentioned above, as illustrated in Fig. 3 Data Pipeline.

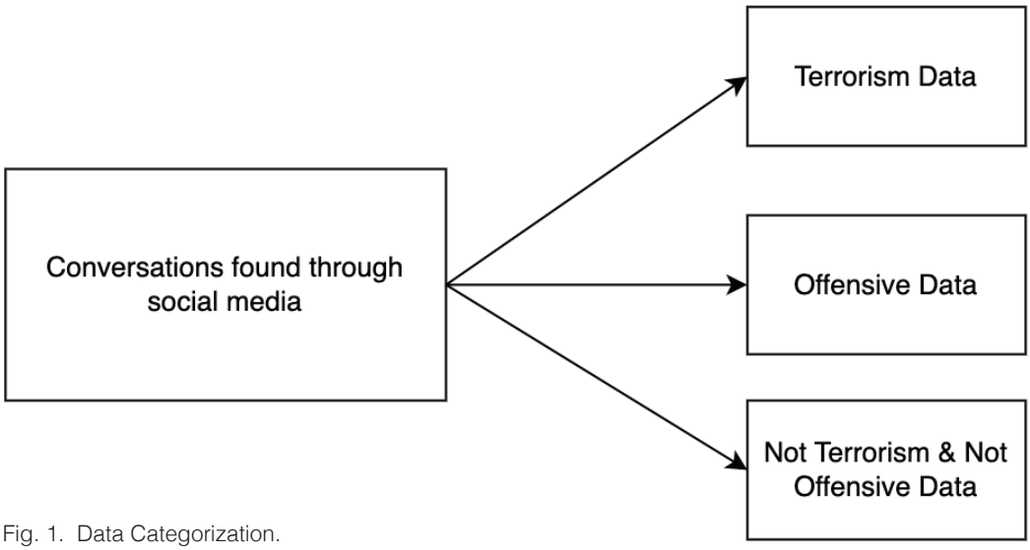

Fig. 1. Data Categorization.

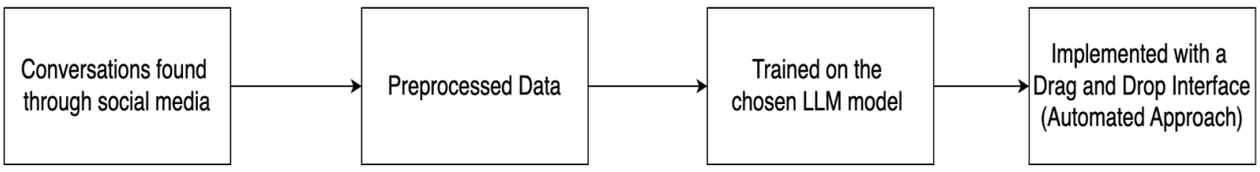

Fig. 2. Proposed Solution.

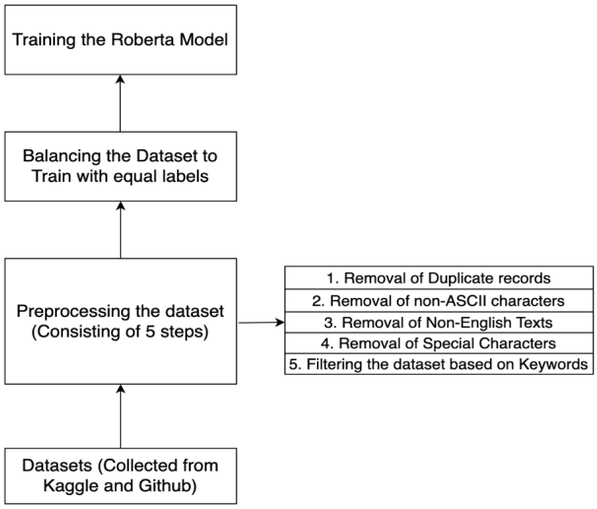

Fig. 3. Data Pipeline.

III. Research Methodology

The research methodology involves a systematic and detailed approach to address all the outlined points based on the assessment of the literature review.

Dataset & AI model: To avoid the dataset being biased and imbalanced, seven datasets from sources such as Kaggle and GitHub that contain text and tweets from Twitter and other messaging platforms have been chosen to train, validate, and test the model. This resolves the issue of bias since the chosen datasets are obtained from different online platforms which affirms that the dataset focuses on other perspectives and ideologies.

To get the best results, one of the current leading AI models needs to be implemented to get optimal results. Based on this criterion and the current trends, we find that LLM models perform excellently in speech and text detection. With the preference for implementing open-source models, the chosen model for this use case is DistilRoBERTa.

DistilRoBERTa: RoBERTa stands for Robustly Optimized BERT approach. It is a variant of the BERT model (Bidirectional Encoder Representation from Transformers), which was developed by Google. DistilRoBERTa is a lightweight model of RoBERTa that mimics the performance, which makes it faster and more efficient due to its less computational footprint. The DistilRoBERTa model is implemented from the transformer library using Hugging Face. Comparatively RoBERTa model performs better than BERT and Distill-BERT, by using DistilRoBERTa, we get the same architecture of the RoBERTa model but with low computational footprint and better performance, making it the optimal choice [1].

The architecture of the DistilRoBERTa model is shown in Fig. 4 DistilRoBERTa Architecture and is explained as follows:

1) RoBERTa for Sequence Classification: It consists of the RoBERTa model and the classification head where the RoBERTa model processes the input sequences (Input Data) and extracts the contextual representations (Tokens), which is then classified by the classification head to produce the logits that is used to determine which label (offensive, terrorism or neither) is to be output as the result. The contextual representations (Tokens) are where data transformation takes places converting input data to unique token ids which means each word is converted to a unique token id. If the same word repeats in each input data sequence, it is given the same id. This helps the model identify complex relationships and enhance its understanding.
2) RoBERTa Model: The RoBERTa model contains the encode for encoding input sequences or embeddings, which are of three layers (The values of the embedding layers are based on the datasets used).
   a) The First embedding layer is the word embeddings, consisting of 50265 unique words; 768 is the size of the vector for each word, and padding is used to make the sequence length equal.
   b) The Second embedding layer is the position embeddings, which consist of 514 positions, 768 is the vector size for each position, and padding is used to make the sequence length equal.
   c) The Third embedding layer is the token type embeddings, consisting of 1 token type, and 768 is the vector size for the token type.

These embeddings are processed in subsequent layers to obtain contextual information.

3) Encoder: The encoder contains six layers, each containing a self-attention mechanism (to help understand the importance of the word and its dependencies) and a feed-forward neural network. This is used to help the model understand the complex relationship between the words from the input sequence.
4) RoBERTa Classification Head: As mentioned above in the RoBERTa for Sequence Classification. This is the last step of the RoBERTa Model, where the logits produced are used as inputs to the last layer which consists of three layers where the first layer is a Linear function followed by a Dropout Layer to avoid overfitting, followed by the final layer which is a Linear function that outputs the label.

IV. Implementation

In the implementation phase, the first process is data collection, which consists of the datasets to create training samples for the model, followed by the RoBERTa model.

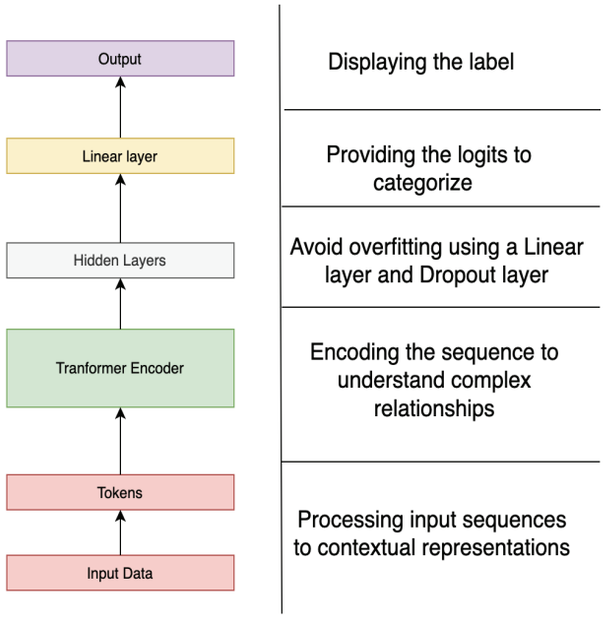

Fig. 4. DistilRoBERTa Architecture.

The datasets used are of two types: Datasets from publicly available sources and Synthetic Data. The research is done on the assumption of identifying chats related to Terrorism, Offensive, and Neither in the English Language only. So, all the chats/texts in the datasets collected are in English. The two main criterions for choosing the datasets were whether it consisted of data that related to Terrorism, Offensive and Neutral and whether the dataset was labelled.

There were seven datasets collected from publicly available sources which are represented in the Table I. Those Data used for Training and Testing & Table II Dataset Description.

The datasets may contain duplicates, empty records or non-English texts and hence need to be preprocessed. Since we only need two columns which are Text and Label, all the other columns will be removed, if present. The information required for training, validating, and testing the model are two columns which are Text and Label where one consists of the text or speech and the other column consists of the associated label that defines which category does an instance fall under (Terrorism, Offensive or Neutral).

Synthetic data is created based on the misclassifications of the model after the training and testing phase which is largely due to the context of the data. For instance, the input data “Plan to bomb the mall” would classify as terrorism while another input data “Did you hear about the bomb blast that occurred few minutes ago” would sometimes be misinterpreted by the model as terrorism instead of neutral since it contains the word ‘bomb’ which is a keyword for terrorism related text. Thus, creating synthetic data by providing different context with the same keyword and re-training helps the model understand and distinguish better.

A. Data Preprocessing

The datasets were preprocessed in 5 steps where the first four steps focus on cleaning the data:

1) Removal of Duplicate/Redundant texts: This involves removing any data (input) that is present more than once.

2) Removal of non-ASCII characters: This involves removing symbols (€, £), Accented letters (é, à) and emojis.
3) Removal of non-English texts: This involves removing texts of languages other than English, present in the dataset.
4) Removal of special characters: This involves removing characters (such as !, $, @, #, /n and others.)
5) Filtering the data from the datasets based on the keywords associated with offensive or terrorism-related texts: This involve identifying data that consists of words that would be classified as offensive or terrorism. The keywords are determined by examining various data that are categorized as offensive or terrorism and choosing the words that fit in that category. For instance, profanity words and discriminatory words will be the keywords used to determine offensive language and words that are related to Violence, attack terms, threats and plots, recruitment and propaganda will be the keywords used for terrorism.

B. Experimental setup

For this research project, Python was used as the programming language, and was implemented on Google Collab. The transformer library from Hugging Face was used to import the DistilRoBERTa pre-trained model.

The hardware resources used is the accessible version of Google Collab consisting of 13 GB RAM as the CPU and Tesla K80 which consists of 2xGK210 chipset, 4992 CUDA cores, 24 GDDR5 and 384-bit Memory interface as the GPU.

The software resources used were Google Collab Notebook, Python, and its libraries, which were TensorFlow, PyTorch, Scikit-Learn, Transformer, Datasets, NumPy, and Pandas.

C. Implementation procedure

1) Firstly, the necessary libraries were imported, which were TensorFlow, PyTorch, Scikit-Learn, Transformer, Datasets, NumPy, and Pandas.
2) Secondly, the datasets were converted to a Data Frame for manipulation and pre-processing, which involved removing

TABLE I
DATASETS USED FOR TRAINING AND TESTING

S.No	Dataset Name	Number of samples	Number of Labels	Source	Author
1	Hate Speech Detection Curated Dataset	440906 records	Offensive : 79305 Non-Offensive: 361594	Kaggle [16]	WENDYELLÉ A. ALBAN NYANTUDRE
2	How ISIS Uses Twitter	17410 records	Terrorism : 17410	Kaggle [17]	Khuram
3	Counter-narratives datasets to fight hate speech	14988 records	Offensive : 14988	GitHub [18]	marcoguerini Marco Guerini
4	GabHateCorpus Dataset	86529 records	Offensive : 11249 Non-Offensive: 75280	GitHub [19]	Mpgiii
5	ABC-news (A Million News Headlines)	1244184 records	Terrorism : 44797 Non-Terrorism : 1199387	Kaggle [20]	Rohit Kulkarni
6	US Mass Shootings May 24 2022	128 records	Terrorism : 128	Kaggle [21]	Zeeshan-ul-hassan Usmani
7	Terrorism And Jihadism Speech Detection	500 records	Terrorism : 250 Non-Terrorism : 250	Kaggle [22]	Haithem Hermessi

TABLE II
DATASET DESCRIPTION

S.No	Dataset Name	Description	Initial Source
1	Hate Speech Detection Curated Dataset	The dataset was obtained by scraping tweets from twitter by identifying whether the tweets and hashtags contained any form of Hate or Offensive speech such as abusive or discriminatory tweets. This dataset was then made available in Kaggle.	Twitter
2	How ISIS Uses Twitter	The dataset was obtained by scraping tweets from twitter by identifying whether the tweets and hashtags promoted terrorism such as the spread of violence and influence of propaganda. This dataset was then made available in Kaggle.	Twitter
3	Counter-narratives datasets to fight hate speech	The dataset was obtained using manual and automated processes derived from various sources such as social media platforms, discussion forums and comment sections which was then filtered based on the selected keywords that would distinguish as Hate and Offensive Speech. This dataset was then made available in GitHub.	Different social media platforms, discussion forums and comment sections (source of dataset not explicitly mentioned)
4	GabHateCorpus Dataset	The dataset was obtained from Gab which is a social media platform that allows users to share content with minimal censorship making it an excellent source for identifying Offensive and Hate speech. The dataset was collected using automated tools and refined manually. This dataset was then made available in GitHub.	Gab
5	ABC-news (A Million News Headlines)	The dataset was collected by scraping a million news headlines from ABC news (Australian Broadcasting Cooperation) that spans a period from 2003 to 2019. This is a good source for identifying Terrorism related events that had taken place. This dataset was then made available in Kaggle.	ABC (Australian Broadcasting Cooperation) news
6	US Mass Shootings May 24 2022	The dataset was collected from publicly available sources that track gun violence and mass shooting incidents reported across the United States. These may include data collected from news articles, government reports or individual contributing to incidents they encountered. This dataset was then made available in Kaggle.	publicly available sources (source of dataset not explicitly mentioned)
7	Terrorism And Jihadism Speech Detection	The dataset was collected from publicly available sources such as social media, discussion forums, comment sections, news articles and other sources where extremist speech can be found. This dataset was then made available in Kaggle.	Social media, discussion forums, comment sections, news articles and other sources (source of dataset not explicitly mentioned)

duplicate texts, non-ASCII characters, non-English texts, special characters, and filtering data based on keywords.

3) Thirdly, the datasets were divided into training, validation, and testing sets and stored in a data dictionary. Then, the tokenizer was initialized with the DistilRoBERTa pertained tokenizer. The tokenizer is used to tokenize the data in the data dictionary, where a data transformation is performed that converts each word or sub-word to unique IDs. Then the compute metrics are initialized to calculate the accuracy, precision, recall, and f1 scores. The output prediction is labeled such that if the output is 1 ,0, or 2, it will output the label as NOT-OFFENSIVE-LANGUAGE & NOT-TERRORISM, OFFENSIVE-LANGUAGE, or TERRORISM, respectively.
4) Lastly, the DistilRoBERTa model is configured with the below hyperparameters:
   - Learning rate: 2e-5;
   - Per device train batch size = 16;
   - Per device eval batch size=16;
   - Weight Decay=0.01;
   - Evaluation strategy ="epoch";
   - Save Strategy = "epoch";
   - Load Best Model at end = True.
     - o Learning rate is the rate at which the model should change based on the error rate.

- Batch size is the number of samples processed before the model is updated (which is the same for both the train batch and evaluation batch).
- Weight Decay is regularization technique to prevent overfitting (prevents the model from learning the noise in the training dataset).
- Evaluation strategy means the model performance is evaluated after every epoch
- Save strategy means the model's checkpoints are saved every epoch
- Load Best model at the end means the most effective version of the model will be chosen which is based on the metric scores such as accuracy.

The dataset is split into training, evaluation and testing datasets where the training dataset is assigned 75% of the dataset, the evaluation dataset is assigned 15% of the dataset and the testing dataset is assigned 10% of the dataset. Then the training and evaluation datasets are initialized, and the training is initiated. After the training is done, the model is evaluated on a testing dataset to determine its accuracy on unknown data as illustrated in Fig. 5 Implementation Process.

Finally, the model is then used with Python Flask, where a drag-and-drop interface is created to upload the conversations stored in text files. The text files are analyzed and classified for each file.

V. Research Findings and Results

The datasets were all combined and balanced such that there were equal numbers of labels for offensive texts, terrorism texts, and neither of them so that the model would not be biased on the label with majority instances but rather treat all the labels equally. The total number of instances (rows) amounted to 211500 instances, and each of the three labels had instances of 70500. The data was split into the train, validation, and test datasets.

The training data comprised of 161797 instances. After the training process is complete the model is evaluated based on the performance accessed by the metric scores as shown in Fig. 6 Performance Metric Formula Calculation, the model's metrics scores were 0.05 for the loss, 99.69% for the accuracy, 99.88% for the ROC score followed by the precision, recall, and f1 scores, which also resulted in 99.69%. The time taken for training the model was 00:57:42 (hours : minutes : seconds). The Validation data comprised of 31725 instances where the model had achieved 0.02 as the loss score. The Test data comprised of 17978 instances and achieved a loss score of 0.031 and an accuracy of 99.53%, ROC score of 99.71%, followed by the precision, recall, and f1 scores also resulted in 99.53%. As seen earlier, the main reason for the model's error or misclassification would largely be due to the context. For instance, if the model was trained with data showcasing attacks on malls, parks and schools as terrorism, the model's interpretation would be to classify an instance as terrorism if it contains malls, parks, or schools even though it was not chosen as a keyword but is due to the dataset being biased by training the model with terrorism activities that occurred in these venues. Looking at the results of this experimentation it could be that the model identified something that is used to classify as terrorism but is neutral or as offensive but is neutral and vice versa, for that case. Since the loss is quite low, this can be rectified and addressed by training with more data which would improve the model's understanding to identify with a clear distinction.

The results of the above experiment as seen in TABLE III Experimentation Results and depicted in Fig. 7 Evaluation Metric scores for both Training and Testing are equal, and Fig. 8 ROC scores for both Training and Testing are equal, shows how the model was evaluated based on the performance metrics which showed that the DistilRoBERTa model achieved excellent results and could categorize accurately with a low loss score. The model was further tested by extracting WhatsApp conversations into a text file format. The text files were uploaded to the drag-drop interface implemented using the Python Flask app, where the DistilRoBERTa model

was used to classify the text files. Below are the illustrations. The training and testing have the same illustrations since the difference of the performance is less than 0.2% which does not reflect any major difference as seen in Fig. 9 Confusion matrix and Classification Report for Training, and Fig. 10 Confusion matrix and Classification Report for Testing.

Three text files were chosen, and the conversations in each of them was related to terrorism, offensive, or neither. The model accurately classified each of the text files with the correct labels as seen in Fig. 11 Drag and Drop Interface to upload files to be categorized, and after Clicking the Classify button the results are display as shown in Fig. 12.

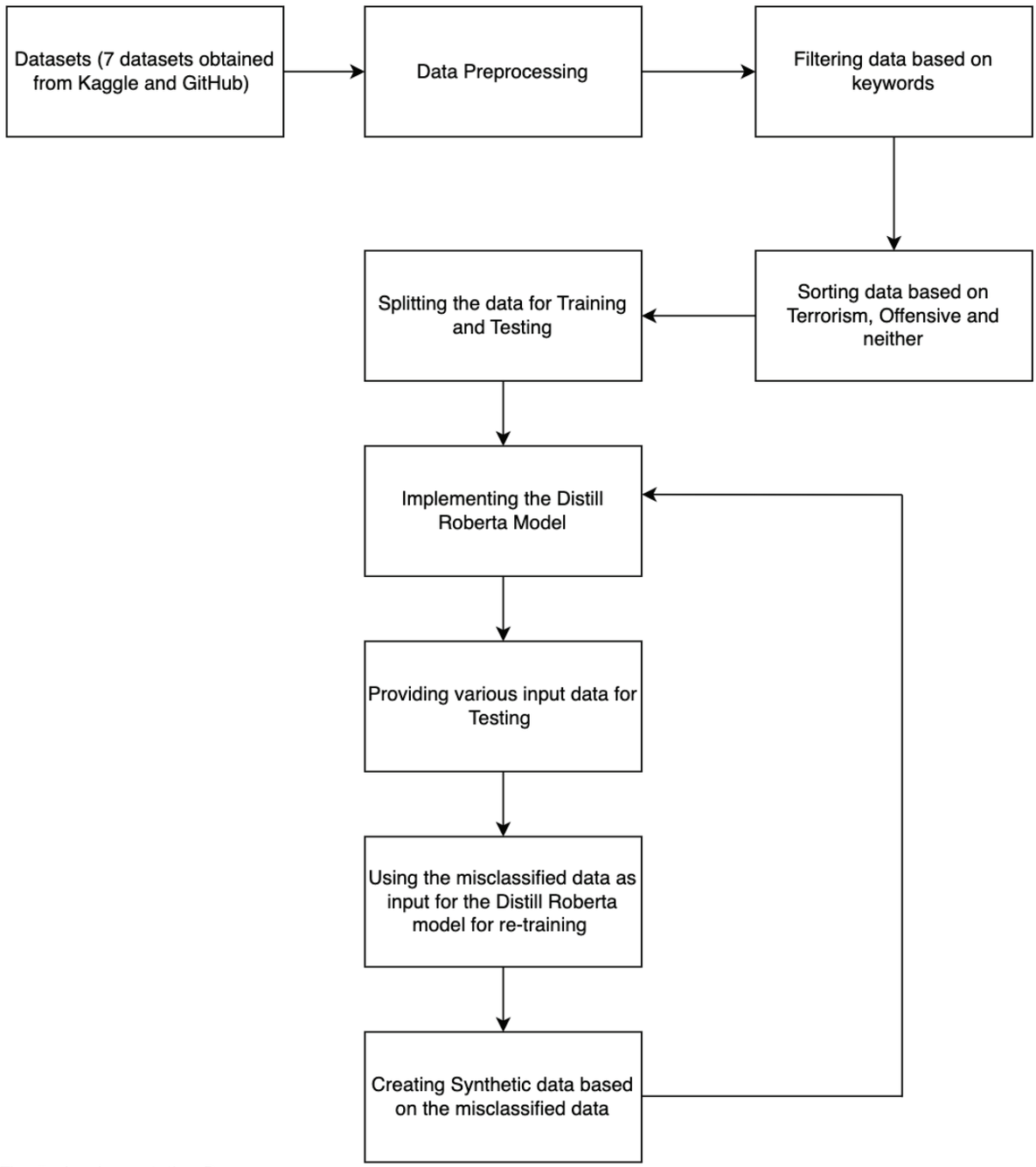

Fig. 5. Implementation Process.

True Positive (TP)	False Positive (FP)
False Negative (FN)	True Negative (TN)

$$\text{Precision} = \frac{TP}{TP + FP} \qquad \text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

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

$$\text{ROC Curve} = \text{plot}(TPR, FPR) \qquad TPR = \frac{TP}{TP + FN} \qquad FPR = \frac{FP}{FP + TN}$$

where TPR is True Positive Rate and FPR is False Positive Rate

Fig. 6. Performance Metric Formula Calculation.

TABLE III
EXPERIMENTATION RESULTS

Datasets	Number of Instances	Loss Score	Accuracy Score	Precision score	Recall score	F1-score	ROC-Curve
Training	161797	0.05	99.69%	99.69%	99.69%	99.69%	99.88%
Validation	31725	0.02					
Testing	17978	0.0031	99.53%	99.53%	99.53%	99.53%	99.71%

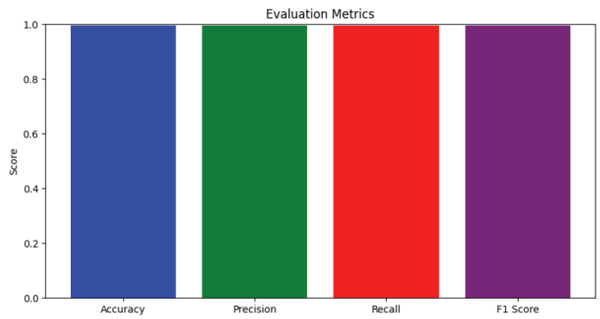

Fig. 7. Evaluation Metric scores for both Training and Testing are equal.

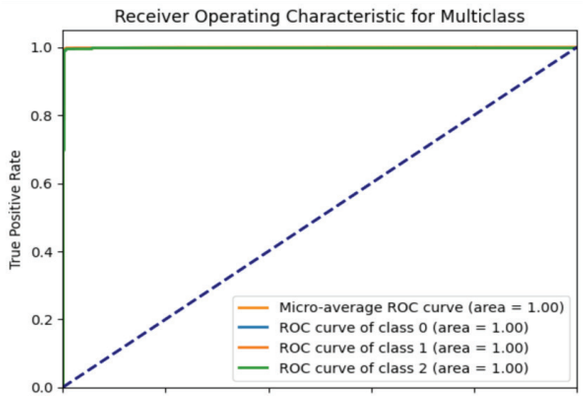

Fig. 8. ROC scores for both Training and Testing are equal.

Confusion Matrix

	0	1	2
0	53784	135	146
1	106	53708	6
2	144	249	53519

0 : Neutal
1 : Offensive
2 : Terrorism

Classification Report

	0	1	2
0	0.99	0.99	0.99
1	0.99	1.0	1.0
2	0.99	0.99	0.99

0 : Neutal
1 : Offensive
2 : Terrorism

Fig. 9. Confusion matrix and Classification Report for Training.

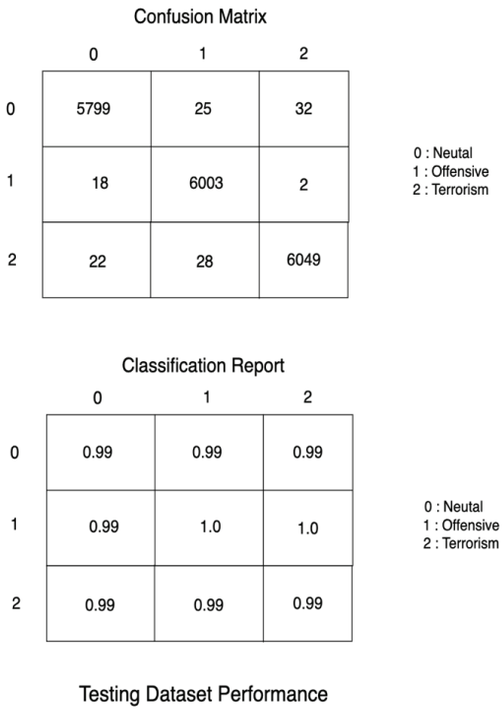

Fig. 10. Confusion matrix and Classification Report for Testing

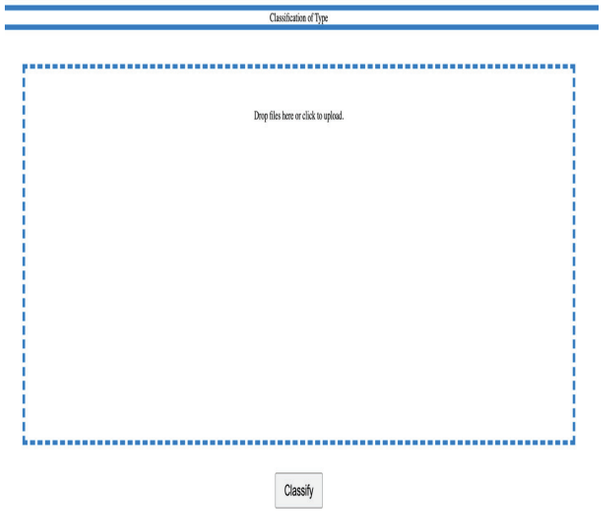

Fig. 11. Drag and Drop Interface to upload files to be categorized

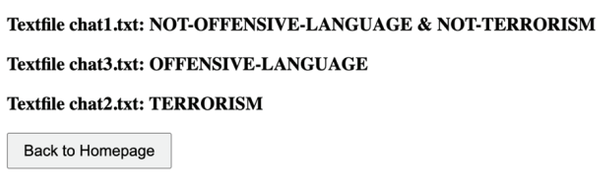

Fig. 12. After Clicking the Classify button the results are display

VI. Limitations, Discussion and Conclusion

Even though the model was trained from different sources such as Twitter and other curated data, the model is slightly biased since the data is filtered based on specific keywords that limit the model in classifying accurately. For instance, a text that is either Terrorism or Offensive are misclassified due to the keywords not identified by the model due to the model not being trained with those keywords. Another factor is the model's limited understanding of the context, where it might classify the data based on the keyword without focusing on the overall context that does not necessarily classify as being related to terrorism or offensive. For instance, if the model considers the death of a hundred people to be considered an act of terrorism, the reason could not necessarily be attributed to terrorism but rather to natural disasters such as earthquakes. This error can be corrected by creating more synthetic data to fine-tune the model by giving similar instances and properly distinguishing what could be categorized as terrorism, offensive, or neither so that the model understands the context better and improves its understanding. Another approach is to obtain high-quality data from sources with access to conversations relating to terrorism or offensive texts since these datasets are not available online for various security reasons. The data collected to train the model may raise ethical concerns, this can be addressed by making it compulsory for the users to accept the code of conduct, terms, and conditions of using the social media platform where communication will be monitored with the aim of preventing Offensive or Terrorism related communication. Another limitation is that the model may not always be accurate and could lead to false positives such as a terrorism related text could be identified as neutral. This can only be addressed by training with more data to improve the model's understanding and distinguish better.

By comparing with the finding of the literature we see that Rajendran et al.[1] and Nithyashree et al.[13] had achieved 98% in accuracy for the Bert model and 95% accuracy for the RoBERTa model respectively which is lower when compared to the

result achieved in this experimentation using DistilRoBERTa which achieved an accuracy of 99%. Even though Mussiraliyeva et al.[7] achieved 99.6% in accuracy using KNN and Gaikwad et al.[5] found three papers achieving 99% in accuracy using AdaBoost, they are not convincingly enough since they have only used one dataset for their experimentation making it biased since they might achieve different results when using different datasets. But in this experimentation, we used different datasets together to train and create a robust model with an aim to classify regardless of the context.

The research project focuses on how datasets from well-known online sources combined with synthetic datasets can be used to train models such as DistilRoBERTa to help automate the task of distinguishing large numbers of conversations in text file format that are taken from messaging apps such as WhatsApp and used to classify as offensive, terrorism or neither. The model was trained on datasets from publicly available sources, and the synthetic data was obtained from the inputs that the model could not classify accurately by creating similar instances and giving more context for the model to understand better. Based on the results, we see that the DistilRoBERTa model was able to classify the labels accurately and achieved a score of 99.53%, indicating that LLMs are far better than many machine and deep learning models, as seen in the Literature review, in accurately classifying offensive and terrorism-related instances. The trained LLM model offers practical applications, for instance, social media providers can leverage the trained model to help identify individuals that spread Offensive speech and take appropriate action by removing the account and if identified as terrorism it can be reported to the law enforcement authorities. Some of the recommendations for future research are implementing the model to use OCR (Optical Character Recognition) to convert images to text or audio to text and categorize them. Another study could focus on converting PDFs to text and using the model to classify the texts. This is a significant challenge since the model can only take limited number of words as input. PDFs may consist of more than 10 pages which increases the number words. This can be resolved by splitting the text into paragraphs or using other techniques to classify data which creates room for research. Lastly, with the robust model that was trained, we create a simple and easy interaction using a drag-and-drop interface created using Python Flask, where text files that contain conversations are uploaded to the interface. The model then classifies the type of category the file belongs to.

The results of the experiment show valuable insights in the capabilities of LLM specifically the DistilRoBERTa model to identify and categorize speech and text in an efficient and effective manner. Social media platforms can leverage the model or further fine tune it with their own data to create an even more robust model to detect terrorism and offensive behavior with the sole purpose of maintaining a safe environment for online communications where users who want to engage and communicate in these platforms will need to follow and adhere to the necessary conduct or face the consequences of their actions.

Funding

This article did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of Interest

Authors declare that they have no conflict of interest.

References

[1] A. Rajendran, V.S. Sahithi, C. Gupta, M. Yadav, S. Ahirrao, K. Kotecha, M. Gaikwad, A. Abraham, N. Ahmed, and S.M. Alhammad, "Detecting extremism on twitter during US capitol riot using deep learning techniques," *IEEE Access*, vol. 10, pp. 133052-133077, 2022.

[2] S. Hussain and P. Mohideen, "Advanced Machine Learning Approach for Detection of Multilinguistic Terror Message to save human Lives," *Journal of Pharmaceutical Negative Results*, pp. 2528-2541, 2023.

[3] O. Sharif, M.M. Hoque, A.S.M. Kayes, R. Nowrozy, and I.H. Sarker, "Detecting suspicious texts using machine learning

techniques," *Applied Sciences*, vol. 10, no. 18, p. 6527, 2020.

[4] A.B. Abhijith and P. Prithvi, "Automated Toxic Chat Synthesis, Reporting and Removing the Chat in Telegram Social Media Using Natural Language Processing Techniques," in *2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, Jan. 2024, pp. 1-7.

[5] M. Gaikwad, S. Ahirrao, S. Phansalkar, and K. Kotecha, "Online extremism detection: A systematic literature review with emphasis on datasets, classification techniques, validation methods, and tools," *IEEE Access*, vol. 9, pp. 48364-48404, 2021.

[6] A. Čepulionytė, J. Toldinas, and B. Lozinskis, "A multilayered preprocessing approach for recognition and classification of malicious social network messages," *Electronics*, vol. 12, no. 18, p. 3785, 2023.

[7] S. Mussiraliyeva, B. Omarov, P. Yoo, and M. Bolatbek, "Applying machine learning techniques for religious extremism detection on online user contents," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 915-934, 2022.

[8] S. Pais et al., "Language-Independent Approaches to Detect Extremism and Collective Radicalisation Online," in *Proc. Conf.*, pp. 7–14, 2020.

[9] F.E.T.H.I. Fkih, T.A.R.E.K. Moulahi, and A.B.D.U.L.A.T.I.F. Alabdulatif, "Machine learning model for offensive speech detection in online social networks slang content," *WSEAS Trans. Inf. Sci. Appl.*, vol. 20, pp. 7-15, 2023.

[10] M. Fahim and S.S. Gokhale, "Identifying social media content supporting proud boys," in *2021 IEEE International Conference on Big Data (Big Data)*, Dec. 2021, pp. 2487-2495.

[11] V. Shirsath, T. Wani, D. Bhakare, P. Lokhande, R. Chavda, and V. Shah, "ChatGuard: A Profanity Classification Approach for Safer Online Conversations," in *2023 1st DMIHER International Conference on Artificial Intelligence in Education and Industry 4.0 (IDICAIEI)*, Nov. 2023, vol. 1, pp. 1-5.

[12] M. Gaikwad, S. Ahirrao, S. Phansalkar, K. Kotecha, and S. Rani, "Multi-Ideology, Multiclass Online Extremism Dataset, and Its Evaluation Using Machine Learning," *Computational Intelligence and Neuroscience*, vol. 2023, no. 1, p. 4563145, 2023.

[13] V. Nithyashree, B.N. Hiremath, L. Vanishree, A. Duvvuri, D.A. Madival, and G. Vidyashree, "Identification of Toxicity in Multimedia Messages for Controlling Cyberbullying on Social Media by Natural Language Processing," in *2022 International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, Oct. 2022, pp. 12-18.

[14] A. Shevtsov, D. Antonakaki, I. Lamprou, P. Pratikakis, and S. Ioannidis, "BotArtist: Twitter bot detection Machine Learning model based on Twitter suspension," *arXiv preprint*, arXiv:2306.00037, 2023.

[15] R. Alshalan and H. Al-Khalifa, "A deep learning approach for automatic hate speech detection in the Saudi twittersphere," *MDPI*, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/23/8614. [Accessed: 20-Jun-2024].

[16] W.A.A. NYANTUDRE, "Hate speech detection curated dataset," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/waalbannyantudre/hate-speech-detection-curated-dataset. [Accessed: 13-Apr-2024].

[17] F. Tribe, "How ISIS uses Twitter," Kaggle, 2019. [Online]. Available: https://www.kaggle.com/datasets/fifthtribe/how-isis-uses-twitter. [Accessed: 13-Apr-2024].

[18] Marcoguerini, "Marcoguerini/Conan: A Repository with several curated datasets of counter-narratives to fight online hate speech," GitHub, 2022. [Online]. Available: https://github.com/marcoguerini/CONAN/tree/master. [Accessed: 13-Apr-2024].

[19] Mpgii, "MPGIII/GAB-Hate: A research project involving trends of hate speech on Social Media Platform Gab," GitHub, 2020. [Online]. Available: https://github.com/mpgiii/gab-hate. [Accessed: 13-Apr-2024].

[20] R. Kulkarni, "A million news headlines," Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/therohk/million-headlines. [Accessed: 13-Apr-2024].

[21] Z. Usmani, "US mass shootings," Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/zusmani/us-mass-shootings-last-50years?select=US%2BMass%2BShootings%2BMay%2B24%2B2022.csv. [Accessed: 13-Apr-2024].

[22] H. Hermessi, "Terrorism and jihadism speech detection," Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/haithemhermessi/terrorism-and-jihadist-speech-detection?select=train.csv. [Accessed: 13-Apr-2024].