International Conference on Recent Trends in Data Science and its Applications
DOI: rp-9788770040723.051

Detection of Virulent Messages Written in Code-Mixed Hindi-English Language

Anita Ramalingam
Department of Computing Technologies
SRM Institute of Science and Technology
Kattankulathur, India
anitaramalingam17@gmail.com

NiravAgarwal
Department of Computing Technologies
SRM Institute of Science and Technology
Kattankulathur, India
na7406@srmist.edu.in

HarshvardhanArvind Singh
Department of Computing Technologies
SRM Institute of Science and Technology
Kattankulathur, India
hs8803@srmist.edu.in

***Abstract*—Social media as a medium of communication has opened countless possibilities for individuals around the globe to express their opinions on various topics ranging from politics to entertainment, thus reducing the gap in the accessibility of information. However, this has also led to social media becoming a hub for spreading hateful messages. These virulent messages spread online result in the rise of cases of cyberbullying, leaving an enduring impact upon the victims. Manually classifying and reporting these messages is a challenging task and has therefore created an interest in the research community to incorporate machine learning-based techniques to ease up the hand-operated process. The study's goal is to play a small auxiliary role in the prevention and ultimate eradication of cyberbullying. The proposed work in this research primarily focuses on the users from the Indian subcontinent by experimenting with Code-Mixed Hindi-English language. This research achieves this by performing the task of developing a Code-Mixed Hindi-English hate speech dataset containing 4014 tweets, out of which 2000 tweets contain hate, using the publically available tweets from the Twitter platform and classified on the basis of a set of guidelines. These languages are employed in the training of the models to create a more reliable hate speech prediction system for the targeted audience. The use of pre-trained multilingual models are explored by fine-tuning on the collected dataset. This research discovered an accuracy of 80.6 on the fine-tuned BERT Multilingual Base model when trained and tested on the combination of all the three languages and an accuracy of 82.2 on theXLM-RoBERTa model upon the same dataset. The goal of the study is to play a small auxiliary role in the prevention and ultimate eradication of cyberbullying.**

Keywords*—Natural Language Processing, Hate Speech Classification, Virulent Messages, Code-Mixed, Dataset Creation, BERT.

I. Introduction

One of the greatest gifts of modern technology to humankind is social media. This gift has truly unlocked the potential of communication by enabling people to discuss and debate on topics and issues that were once considered off-limits due to a number of reasons, like social stigma or regulations prohibiting free speech. In that way, social media has empowered individuals across the globe with the opportunity to express their opinions, views, criticism and experiences on a wide variety of topics, ranging from politics to entertainment. One such popular social-media platform in today's time is Twitter. In the past few years, Twitter has been extensively used by politicians, media houses, organizations, and individuals to share information such as facts, reports and updates regarding the events taking place in the real world, and correspondingly has been used to debate on the same.

Unfortunately, there is always a flip side to the coin. It cannot be denied that online social media platforms allow users to adopt personas and pseudonyms without providing any accountability for what they say. This lack of restrictions and the faceless nature of the internet have made it easy for people to post content online that can be deemed as hate speech. Generally, Hate speech is conveyed as the depiction of communication that is hateful, demeaning, intolerant, and in some way is degrading and inharmonious. There is no proper definition of the virulent message but in many instances accepted meaning deals with communication in speech, behavior, or writing, remarks which are insulting or abusive concerning an individual or a group of individuals, either directly or indirectly. Rise in spread of hate speech has contributed to a rise in reports of cyberbullying, wherein certain individuals and groups of individuals have emerged with the purpose of spreading misinformation and hate online by attempting to silence dissenting voices and harassing them through repeated threats and abuse. This results in a damaging and usually enduring impact on cyberbullying victims, which may lead to low self-esteem, depression, and other mental health issues. The situation has further deteriorated due to the effects of the Covid-19 pandemic. The Covid-19 pandemic has forced the majority of the globe to shift to online mode of education, work and other interactions, exposing more individuals to the kind of hate speech mentioned earlier, especially children who have moved to the online medium for their educational requirements.

Classifying such hate speech is inherently a challenging task since it is difficult to develop universally accepted guidelines and definitions for hate speech. The precise description of what can be cited as hate speech varies widely based on the language used to convey the message, its context, and the time and location when the message was given, making it an extremely subjective issue. It is hence stated that some may disagree with what others find offensive. Therefore, it has sparked a conversation in the research community, drawing an active interest in the field of hate speech classification, which is considered to be one of the most prominent real-life use-cases of sentiment analysis in Natural Language Processing (NLP). As mentioned earlier, the task of hate speech classification is highly subjective, and one of the factors affecting it is language. Therefore, it is essential to understand the language used to convey a message. When discussing the Indian subcontinent, one of the prominently used languages is Hindi, which is approximately 520 million people speak. It is observed that with respect to the language utilized in social media conversations, the use of a Code-Mixed

DOI: rp-9788770040723.051

language in Roman (Latin) script is favored over the use of native language structure. The same holds valid for the Hindi language, wherein the Code-Mixed Hindi-English is commonly used in online conversations over the conventional Devanagari Hindi structure. Thus, this study focuses on Code-Mixed Hindi-English, and Hindi languages along with the English language.

This research deals with the procedure of gathering publically available tweets from the Twitter platform to create a reliable Code-Mixed Hindi-English dataset. This dataset can then be used to train the state-of-the-art deep-learning model(s) to identify hate text written in the same language.

Over the course of this research, a lot of challenges and limitations were encountered while trying to identify virulent texts from a given set of texts. Some of the identified challenges are as follows:

- First, due to the code-mixed nature of Hindi, there is a lack of pre-existing datasets for Hindi. This required us to create our own dataset from scratch, which was challenging. Our dataset was limited in size and scope, so we had to combine multiple datasets into one. This posed a bit of a challenge because we needed to ensure that the data was not overlapping or conflicting with other sources.
- In addition to this, we also experienced difficulty with the NLP libraries. Although Indic NLP and iNLTK are available for pre-processing Hindi Language, the same doesn't work effectively with Hindi Code-Mix. We were required to do some hand coding and development in order to get it working properly.
- Additionally, strong biases against particular religions like "Islam" and "Hinduism", and particular communities like the LGBTQ community are usually observed and needs to be eliminated, so as to avoid creating a biased model.
- Due to the heavy computation required to train deep neural network models, a lot of computational power was required. It prompted the use of Graphics Processing Unit (GPU) to accelerate the training process by utilizing the principle of parallel computing.
- Also, a few individuals are aware of such techniques, and actively try to evade detection by using slangs or misspelled words. It also needs to be taken into account.

The remainder of the paper is organized as follows. Section 2 describes the related work and section 3 explains about the dataset creation. The proposed work is described in section 4 and results are discussed in section 5. Section 6 explains the conclusions and future work.

II. Related Work

In this section the works done on the hate speech classification in English language and and code mixed language in Indian subcontinent are discussed.

A. Works Done on the Hate Speech Classification in English Language

In recent times, with the advancement of computation technologies and the introduction of more advanced machine learning techniques, there has been an increased interest in the field of NLP to classify textual data into various classes. One of the most prominent classes in this is that of hate speech. A considerable amount of research work has been done in the past decade to understand and create highly reliable and sophisticated machine-learning-based models to classify the data as hate speech or not.

A major contributing factor in this task was done by Waseem Z. and Hovy D. [1] in 2016 when they analyzed a publicly available corpus of 16k tweets to classify hate speech based on various criteria for the English language. Furthermore, the authors examined the influence of numerous extra-linguistic features in accordance with the character n-gram. The authors collected tweets over a span of two months and created an annotated corpus of 16.9k tweets containing 1.9k racist and 3.3k sexist comments. The authors further explored the data by analyzing diverse aspects such as the demographic, geographical and lexical distribution. Lastly, the authors derived a list of criteria helpful to identify racist and sexist comments.

In the year 2017, Ji Ho Park, et al. [2] used the same dataset to suggest one-step and two-step classification of hate speech. The authors received an F-measure of 0.827 on the HybridCNN in one- step and an F-measure of 0.824 on Logistic Regression (LR) in two-steps. The authors also implemented Convolutional Neural Network (CNN) models on character, word and hybrid level. Apart from the three CNN models, SVM and FastText were also used to observe the performance for comparative purposes. The same dataset was further used by Pitsilis G.K, et al. [3] in 2018 to experiment on various features related to the user, which includes the past mentions of racist and sexist comments by users on an ensemble of Recurrent Neural Network classifiers. The authors also used word frequency vectors derived from the textual content of tweets. Furthermore, the authors compared performances with single classifiers along with the performances noted by past researchers. The results proved to be comparable and sometimes even better than the state-of-the-art algorithms at the time.

MacAvaney S, et.al. [4] in 2019 in their research discussed the technical and practical challenges when it comes to hate-speech detection. This comprised challenges such as the limited availability of data for training and the discussion about the non-existence of a universal definition for hate speech. The authors proposed a multiview SVM approach that was able to achieve performances similar to the state-of-the-art architecture whilst having a more straightforward, easy to interpret decision-making structure. The authors also used an array of datasets such as the Waseem A [5] and B dataset [6], Stormfront dataset [7], TRAC dataset [8] etc.

Using the HASOC 2019 data, GyörgyKovács, et. al. [9] further explored the challenges involved in identifying hate speech on Social Media and have explored in their study various opportunities, such as leveraging unlabeled data, similarly labelled corpora, as well as the use of novel models. The authors proposed a deep neural network

DOI: rp-9788770040723.051
comprising both the recurrent and the convolutional layers, making the use of CNN - Long-Short Term Memory (LSTM) architecture. The authors also deployed the use of the FastText classifier and the pre-trained RoBERTa model to evaluate its result. Lastly, the authors also experimented with an ensemble of different models to assess and compare the performance with single systems.

Marian-Andrei Rizoiu, et al. [10] in 2019 used the dataset provided by Waseem and Davidson to train a state-of-art model to classify the texts present. It was achieved by using a deep neural network along with transfer learning to create a model that can create word and sentence embedding specific to the task of hate speech classification. The authors further discuss using the models to generate a two-dimensional text visualization process termed Map of Hate. This process can separate different kinds of hate speech and try to illustrate what makes those texts dangerous for the users. The authors aimed to propose models that can reduce the manual work performed by human moderators on chat platforms and automate the same process.

It is clearly evident that a great deal of research has been done for the identification of hate speech in the English language with the use of state-of-art classification techniques. This has further sparked the interest to experiment and extend the same to other languages around the globe. One of the most prominent currently spoken on social media in the Indian Subcontinent is Code-Mixed Hindi-English.

B. Works Done on the Hate Speech Classification in Code Mixed Language in the Indian Subcontinent

In the year 2016, Prabhu A., et al. [11] performed sentiment analysis on Code-Mixed Hindi-English and proposed an annotated dataset for the same. Further, the authors introduced sub-word level representations in Subword-LSTM [12] architecture as compared to the traditional approach of character-level or word-level representations. This proved useful even in the case of highly noisy data, i.e. with a lot of misspellings and/or other mistakes. The authors were able to achieve higher accuracy by the margin of 4 % to 5 % as compared to the traditional approaches.

A major contributor in the classification of the code mixed Hindi-English language is the work done by P. Mathur, et al. [13] in 2018 wherein they created a dataset for Code-Mixed Hindi-English language and used transfer learning coupled with multiple feature inputs to identify hate speech. It was successful using the concept of Multi-Input Multi-Channel Transfer Learning Based Model (MIMCT) that was used to detect hate or abusive content. The authors proposed the Hinglish (Hindi-English) Offensive Tweet (HOT) dataset in the study. A comparison was also shown with the baseline supervised classifiers and transfer learning-based CNN - LSTM models. The prescribed MIMCT model by the authors contained two primary components, which included primary and secondary inputs, and CNN - LSTM binary neural network. The author included the Sentiment Score (SS), LIWC Features, and Profanity Vector in the primary and secondary inputs. Here LIWC features included linguistic statistics, current concerns, spoken categories, textual categories, psychological processes, and grammatical structures.

In the same year 2018, AdityaBohra, et al [14] made a significant contribution by presenting a Code-Mixed Hindi-English dataset with word level annotation. The authors examined the difficulties related to recognizing hate speech in code-mixed texts. The authors further presented a supervised classification system that included numerous components such as character level, word level and lexicon-based features. The authors extracted features such as character N-gram, word N-gram, punctuations, negative words, and lexicon. The authors tested the support vector machine-based classifier on each feature separately and on all combined features. The authors achieved an accuracy of 71.7% using the support vector machine-based classifier. The authors also compared the results with the random forest classifier.

In the year 2018, Kamble S., et al. [15] observed that using domain-specific embeddings results in an improved representation of target groups mentioned in the Code-Mixed Hindi-English dataset by Bohra A., [14]. The models proposed by the authors resulted in an F-score that was 12 % higher as compared to the F-score achieved in the past using statistical classifiers. Instead of utilizing the pre-trained word-embeddings, the authors trained word-embeddings on an enormous corpus of pertinent code-mixed texts. The deep learning models proposed by the authors included the CNN-1D,LSTM and BiLSTM models. The authors also used characteristics that included features like number of tweets, number of timelines extracted, number of retweets, the total number of words, size of vocabulary, and percentage of Hindi words per tweet.

Santosh T.Y.S.S, et al. [16] in 2019 further explored various techniques such as attention based on phonemic sub-words on two architectures that are hierarchical LSTM and sub-word level LSTM. This was accomplished with the help of the publicly available code-mixed dataset. The authors also compared the results with a support vector machine-based classifier and random forest classifier. The hierarchical LSTM model with attention based on phonemic sub-words contained the embedding layer, the syllable encoder, and the word encoder, along with word attention and an output layer.

Sreelakshmia K., et al. [17] in 2020 used Facebook's pre-trained word embedding library, fastText to represent 10000 data samples collected from different sources as hate and non-hate. The authors carried out the experiment for the Code-Mixed Hindi-English language. The authors also compared the results with the word2vec and doc2vec features. With this, the authors achieved an accuracy of 85.81 % by using the proposed methodology with a Support Vector Machine (SVM)-Radial Basis Function (RBF) classifier.

By the end of the same year 2020, Vashistha N., et al. [18] combined multiple datasets available for the English and the Hindi language along with the dataset available for the Code-Mixed Hindi-English to test on a variety of deep neural networks. The author firstly built a baseline model and then used several optimization strategies to increase the

DOI: rp-9788770040723.051

model's performance. The authors developed a tool that detects and rates a given comment with an effective metric in near-real-time and uses the same feedback to further re-train the model, following which the authors achieved a competitive performance score. In two languages, English and Hindi, the authors demonstrated the efficiency of their multilingual model.

The frequency of harsh language on social media motivated Gaikwad S., et al. [19] in the year 2021 to develop techniques that could detect such content automatically. Apart from a few oddities, the majority of research studies have focused on the English language; hence the dataset termed MOLD, which stands for Marathi Offensive Language Dataset, was created to address this issue. It's the first of the kind dataset created for the Marathi language, and it's opened up a whole new field of study for low-resource Indian subcontinental languages. The authors used state-of-the-art cross-lingual transformers to explore machine learning models, including zero-short and other transfer learning experiments, using existing Bengali, English, and Hindi data.

In the year 2021, Sazzed S. [20] proposed the study taking into account the Indian regional languages used in the Indian Subcontinent. The author created a Bengali language corpus of 3000 comments divided into hate and non-hate, having a ratio of 1:1. Furthermore, the author tested the proposed dataset on various machine learning and deep learning classifiers such as support vector machines to detect abusive comments.

In the year 2022, Arushi S., et al. [21] proposed a study that focuses on identifying hate speech in Hindi–English Code-Switched languages. The authors' research entails experimenting with transformation strategies to obtain an accurate text representation. The authors constructed 'MoH', which stands for Map Only Hindi. The term MoH conveys 'love' in the Hindi language. The proposed 'MoH' is a Hindi language pipeline, which consists of language identification that assists with the process of transliteration of Roman formatted Hindi to Devanagari Hindi language using a knowledge base of Roman Hindi terms. Furthermore, the authors fine-tuned Multilingual Bert and, as an extension of that, the [22] Multilingual Representations for Indian Languages (MuRIL) model, as a part of the 'MoH' pipeline.

Based on the research and study performed during the work process for this research, a list of demerits of past implementations was compiled to better understand the requirements and need for this research. In today's time, most of the communications that are happening on social media platforms online are carried in the code-mixed version of the languages. Therefore, it is crucial to develop a system that can process these code-mixed languages. A number of models have been proposed in the past that are created by training on the native alphabet format of the languages. However, these models cannot be applied directly to process code-mixed languages due to their limitations on language modeling and translation. Hate speech classification is a crucial task to be performed in today's era. Therefore, it should not be taken lightly. One of the most noteworthy things to consider when dealing with hate speech is the context in which it was said. There are many resources and publicly available datasets for the English language that contains annotations to classify a sequence as hate or not hate, and which categorize the hate sequences into multiple categories like abusive, offensive, racist, sexist, ethnic, etc. Unfortunately, the same isn't available in abundance in other languages, especially in the code-mixed languages.

A majority of model proposed in the past were monolingual, primarily focusing on the English Language. Whilst, this research mainly concentrates on the vast majority of users from the Indian subcontinent, who speak Hindi and, as an expansion, Code-Mixed Hindi. In this research, various machine learning models are tested alongside with advance d deep learning models on the English, Hindi and Code-Mixed Hindi-English Dataset. This study also includes the process of collecting and creating a new dataset for the Code-Mixed Hindi-English Dataset, by establishing a set of guidelines and rules that can be followed to annotate Code-Mixed Hindi-English texts as hate speech or non-hate speech.

III. Dataset Creation

A wealthy amount of pre-annotated data is available for the task of hate speech classification when it comes to the English language and also the Devanagari Hindi language. The same doesn't hold true for Code-Mixed languages such as Code-Mixed Hindi-English. Hence, this prompts the need to gather and prepare datasets for Code-Mixed languages. This section details the complete strategy followed during the span of the creation of the presented Code-Mixed Hindi-English dataset. This strategy involved scraping publicly available tweets from the Twitter platform, manually classifying the dataset based on a specific set of rules and guidelines set by the annotators before the commencement of the data annotation stage, and lastly, documenting the statistics of the combined dataset.

A. Data Collection

As stated earlier, the dataset proposed in this study is created from the publically available tweets on Twitter. The task of scrapping the tweets can be accomplished using Python libraries that access the official Twitter API, for example, the open-source library, Tweepy[18]. Apart from the Tweepy, a few more additional Python libraries are available that provide similar functionality. To build the Twitter dataset for this study, the Snscrape Python library was used. Snscrape is a scraper for social networking services (SNS), providing the feature to scrape data from a plethora of social media platforms, including Facebook, Instagram and Twitter.

A set of keywords and hashtags was chosen, accounting for various current and past events, topics, and discussions occurring on Twitter. The aforementioned set of keywords and hashtags was created with keeping the objective of fetching as many Code-Mixed Hindi-English language tweets as possible in mind. This set was then used with Snscrape to scrape 25,303 tweets from Twitter.

It included the original tweet text, the username of the

person who posted the tweet, the timestamp (date and time) of the tweet, and the tweet's unique tweet ID.

B. Data Annotation

The step of classifying the collected data into the hate and the non-hate class was carried out manually by reading and examining all the tweets. This process was performed by bilingual authors who are well versed in both the English and Hindi language. In order to avoid any type of bias from the authors, a set of policies and rules were debated and agreed on before initiating the task of manual classification.

Firstly, apart from the original tweet text, other data values such as the author of the tweet, the tweet ID, and the timestamps were hidden during the course of manual classification. It was done to avoid any bias that may arise based on the tweet's author. Furthermore, the data was shuffled before splitting among the authors to allow each author to get tweets based on diverse keywords and hashtags instead of just a few. As discussed previously in the paper, it is hard to define a standard definition of hate speech, so to acquire a uniform approach to classification by the authors, the following set of guidelines was followed.

1. Any tweet containing severe profanity, abuse, threat, offensive remarks, personal insult, harassment, promotion of violence and harm were classified as hate.
2. Any tweet not following in the above category was classified as non-hate.
3. Tweets containing texts written with some sense of sarcasm were handled on a per tweet basis, where tweets without any malign intent or explicit depiction of hate were marked as non-hate, whilst others were labelled as hate.
4. A tweet containing criticism about any community, group or person was classified as non-hate as long as the tweet didn't include any offensive remark, use of extreme profanity or the intention to denigrate the targeted individual or group of individuals.
5. A maximum threshold was set for the number of words allowed in a tweet belonging to the English and Hindi (Devanagari Format) language with respect to the total number of words in the same tweet. If the use of such words were within the threshold, the tweet was considered for classification; otherwise, the tweet was removed from the data.
6. Tweets not following the above guidelines were considered garbage tweets.
7. Tweets that were challenging to classify were flagged for reexamination and were reviewed later by all the authors.

Once the authors completed the initial data classification stage, the tweets marked as garbage were removed, and the rest of the data was compiled. The authors reviewed and weighed on the classification performed by their peers, which was later used to calculate the inter-rater reliability score. During this step, thorough discussion was conducted on all tweets previously flagged for reexamination.

C. Dataset Statistics

The dataset produced from the manual classification task contains 2000 hateful tweets out of a total of 4,014 tweets, resulting in the percentage of hateful tweets being 49.82%. Table 1 shows the hate class distribution. The example data in the dataset are given in Table 2. As stated earlier, the inter-rater reliability (IRR) score was tallied to gauge the consistency of the data classification performed by the authors, using the Cohen's Kappa coefficient. The resulting value of 0.964 indicated a high level of agreement.

TABLE 1: HATE CLASS DISTRIBUTION

Class	Data Entries
Non - Hate	2014 (50.18%)
Hate	2000 (49.82%)
Total	4014

TABLE 2: Dataset Values (Censored)

Text	Class
Yaaraajkalkisicheezmeinmaan hi nahilagta, bas alas alasrehtahaihar time	Non - Hate
Waqtlgjaata h duniyakobsbaatsamjhne me, Zamanalgjaataunhejazbaatsamjhne me #khushi https://t.co/zLOqhSaqK8	Non – Hate
@karanjohar Are m*dherch*d hi*rasa*a tu hi*ra m*adherch*d teri sari movie kaboycotttkrunga hi*re kipaidaish m*adherch*d ha*ala	Hate
@TwitterIndia tum ga*du follower kyuapnega*d me ghusalete ho	Hate

IV. MODEL ARCHITECTURE

This section will discuss the various procedural steps that were followed to evaluate the created dataset by utilizing it to fine-tune an array of pre-trained deep learning models. It involved the process of cleaning the dataset by removing all the not required data, performing a basic exploratory data analysis on the dataset, fine-tuning the hyper parameters during the period of model re-training and evaluation.

A. Data Pre-Processing

To process the text data for use in a deep-learning model, first, it was necessary to clean the data and remove all extraneous information. This step was accomplished by removing all nonessential columns from the dataset (tweet ID, tweet author, tweet timestamp). Furthermore, publicly available text processing Python libraries were used to streamline the cleaning and preprocessing of the tweets' textual data. This set of tools includes the Natural Language Toolkit (NLTK) and Ekphrasis library. With these tools, the tweets were processed by removing stopwords (For English Words present in the tweet), unpacking the hashtags and contractions, annotations of URLs, users, emoticons, date, time and hashtags. Table 3 shows the example data before and after cleaning.

TABLE 3: DATA EXAMPLE BEFORE AND AFTER CLEANING

Data Cleaning	Text
Before	@ANI Khudkegharmeinbijlinahiaurchaledusrokora ahdikhane #powergrid failure
After	user khudkegharmeinbijlinahiaurchaledusrokora ahdikhane power grid failure

B. Pre-trained Deep Learning Models

DOI: rp-9788770040723.051

The resulting preprocessed dataset was then fed into a range of pre-

C. trained deep learning models, including the Multilingual Bert and XLM- RoBERTa [19, 20].

i. Multilingual BERT - BERT stands for Bidirectional Encoder Representations from Transformers. It was first introduced by Google AI Language. Bidirectional means that the model can read text from both left-to-right and right-to-left. One of its iterations is Multilingual BERT, a transformers model pre-trained on a large set of multilingual data in a self-supervised manner, enabling comprehension of multiple languages.

ii. XLM - RoBERTa - RoBERTa is a robustly optimised method for pretraining self-supervised NLP systems proposed by Facebook AI. These models can be tuned for various tasks such as document similarity and classification. One variety of this model is XLM - RoBERTa (XLM-R). XLM denotes a Cross-lingual Language Model [21]. XLM-R is a multilingual model pre-trained on 2.5 TB of filtered CommonCrawl data in 100 languages.

Several renditions of XLM-RoBERTa are available, which includes XLM-RoBERTa-base and XLM-RoBERTa-large.

For the purposes of this study, bert-base-multilingual-cased model and xlm-roberta-base model were selected from the Hugging Face platform.

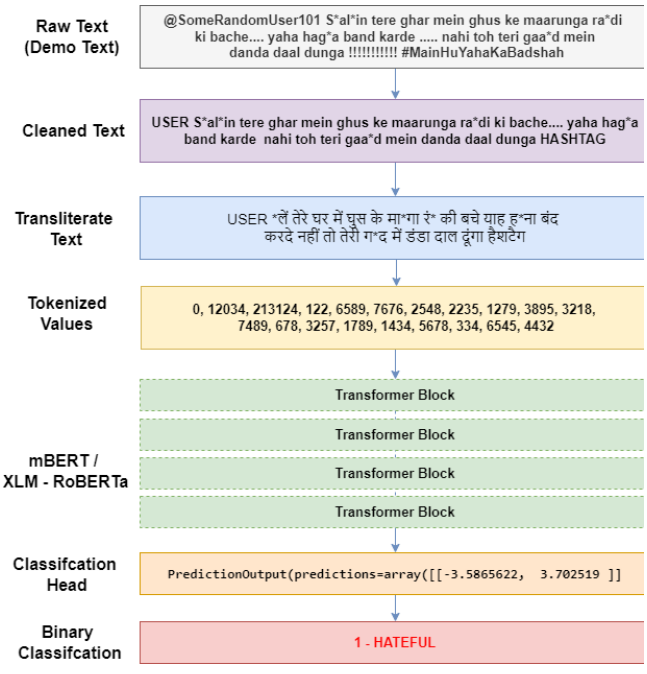

Fig. 1. System Architecture

Fig. 1 shows the system architecture which tells the step by step procedure.

V. Results and Discussion

This section will cover the performance of the machine-learning and deep-learning algorithms used during this study. The results are presented in a tabulated order, and noteworthy observations are drawn from the above-stated results of the research. This section also details the various parameter and hyper-parameters used during the training phase of this study.

A. Basic Machine Learning Classifiers

The feature generated from the data for all three languages were individually divided into a train and test set of 70% and 30%, respectively. The extracted features were thenpassed on as an input to an array of machine-learning-based classification techniques. Different parameters such as the number of estimators or the penalty were experimented for the various classifiers.

- Mainly, the models were trained with a K-Folds cross-validator having a value 5 for the number of folds and also, using the method of grid search.
- For the Logistic Regression (LR) classifier, the penalty was chosen as 'l2', an inverse of regularization strength of 0.02, and class weight was chosen as 'balanced' to give equal weightage to both the classes.
- For the Support Vector Machine (SVM) classifier, the regularization parameter was set to 0.01, and the kernel was chosen as 'linear' because of the number of features used in the input.
- For the Random Forest Classifier (RFC), depending on the language, the number of decision trees varied from 120 to 150. Also, the 'gini' criterion was used.

Tables 4, 5 and 6 show the machine learning (ML) Classifiers' accuracy, precision, recall and F1-score for English, Hindi, and Code-mixed Hindi-English languages.

TABLE 4: ML Classifier Results for English Language

Classifier	Accuracy	Precision	Recall	F1-Score
LR Classifier	0.779	0.774	0.775	0.774
SVM Classifier	0.798	0.794	0.790	0.792
RFC Classifier	0.821	0.816	0.819	0.817

TABLE 5: ML Classifier Results for Hindi Language

Classifier	Accuracy	Precision	Recall	F1-Score
LR Classifier	0.785	0.784	0.784	0.784
SVM Classifier	0.814	0.807	0.809	0.808
RFC Classifier	0.834	0.832	0.831	0.831

From the results mentioned above, several observations can be made. Broadly, the Random Forest Classifier (RFC) model appears to yield the best results in all the languages.

TABLE 6: ML Classifier Results for Code-Mixed Hindi-English Language

Classifier	Accuracy	Precision	Recall	F1-Score
LR Classifier	0.712	0.710	0.711	0.710
SVM Classifier	0.735	0.734	0.734	0.734
RFC Classifier	0.753	0.750	0.751	0.750

The reason behind this may be related to the fact that RFC operates on multiple decision trees reducing any bias or over-fitting that may occur when working with only one decision tree or a single procedure classifier. Also, important to note that the results for the English and the Hindi language are considerably better than that for the

Code-Mixed Hindi-English language. This can be substituted to the fact that the more robust techniques and tools for data cleaning and pre-processing are available when it comes to the English and Hindi languages. One other reason that may contribute to this is that the Code-Mixed version of Hindi doesn't have a fixed vocabulary, and due to variation in the method of writing code-mixed language, a single word in Devanagari Hindi may have several spellings in Code-Mixed (Roman) Hindi, resulting in a vast vocabulary and thus, reducing the performance across the board. The results are assumed as the baseline performance achieved on the dataset.

B. Fine-Tuned Deep Learning Models

This section presents the results of the experiments performed on the pre-trained deep learning models using the created datasets. Since the dataset was tokenized using the sub-word tokenizer available with the respective models, the dataset was first translated to Hindi (Devanagari) Language using the Google translate library available for Python, and then it was tokenized. The dataset was tokenized with a maxed length of 256 and padding based on the same. It was further encoded using the transformer function available on the above-mentioned platform. The encoded dataset values with corresponding class labels were then transferred as an input to the models.

A set of hyper-parameters was tested during the model fine-tuning period; this includes a learning rate of 2e-5, batch size of 16, weight decay of 0.01, and train-validation-test split of 70-10-20. The models were terminated with a classification head to produce the binary hate speech classification results.

Tables 7, 8 and 9 show the deep learning (DL) Classifiers' accuracy, precision, recall and F1-score for English, Hindi, and Code-mixed Hindi-English languages.

TABLE 7: DL MODEL RESULTS FOR ENGLISH LANGUAGE

Model	Accuracy	Precision	Recall	F1-Score
mBERTModel	0.834	0.830	0.831	0.830
XLM-RoBERTaModel	0.851	0.848	0.849	0.848

TABLE 8: DL MODEL RESULTS FOR HINDI LANGUAGE

Model	Accuracy	Precision	Recall	F1-Score
mBERTModel	0.857	0.856	0.856	0.856
XLM-RoBERTaModel	0.879	0.876	0.878	0.877

TABLE 9: DL MODEL RESULTS FOR CODE-MIXED HINDI-ENGLISH LANGUAGE

Model	Accuracy	Precision	Recall	F1-Score
mBERTModel	0.771	0.770	0.769	0.769
XLM-RoBERTaModel	0.784	0.782	0.783	0.782

A considerable improvement has been observed when compared to the performance delivered by the baseline machine-learning-based classifiers. There have been significant improvements for all the languages separately. But, when observing the results acquired from the models that were fine-tuned on all the datasets combined, the evaluation score appears to be an average of the results of the separate models. This further solidifies the theory that has been proposed regarding one of the drawbacks of the multilingual model approach. Also, it has been observed that the XLM-RoBERTamodel performed better as compared to the mBERTmodel across all the languages. This may be because of the use of Byte-Pair Encoding (BPE) used by XLM-RoBERTa which allows it to have an increased shared vocabulary between languages, which in the case of this research, benefits the Devanagari Hindi and the transliterated Code-Mixed Hindi-English languages.

VI. CONCLUSIONS AND FUTURE WORK

As iterated earlier, this study aimed to play a small auxiliary role in the prevention and ultimate eradication of cyberbullying. The first step towards that direction is the creation of state-of-the-art classifications models that can classify hate messages. The recent interest in this field of research has yielded some promising results, especially in the realm of English language with enormous amounts of data available for training. But, as demonstrated in recent research efforts, this success can be taken further into the domain of other languages. It can thus be applied to a broader audience who speak different languages. The study also emphasizes the fact that the communication that is taking place in today's time on social media forums and platforms usually contain a mix of various languages, some of which are code-mixed versions of English and the native language of the region.

This study delivered a hate speech (virulent text) classification dataset for Code-MixedHindi-English language with 4,014 values, with an even distribution of hate and non-hate entries manually classified from a scraped Twitter corpus of 25,303 tweets. Furthermore, this study also proposed a set of policies and rules that can be followed to classify hate data. The guidelines followed in this study can also be used in future research to create more sophisticated hate-speech datasets. The guidelines can also be employed to further explore the data by dividing the hate speech class into multiple classes, such as abuse, threat, insult, and harassment. This study also attempted to fine-tune two deep-learning models, namely, multilingual BERT and XLM - RoBERTa, on the created dataset, providing models that can effectively identify texts containing hate messages.

More in-depth work is still required to improve the models further and deploy the studied techniques into real-life applications. With issues, such as active avoidance (user trying to evade hate speech detection by using misspelt words or slang) and community bias (race, religion, gender, occupation, etc.) continuing to affect the performance of classical hate speech flagging systems, the need to create more reliable automatic machine learning-based techniques is crucial in the forthcoming times.

REFERENCES

[1] Z.Waseem, and D.Hovy, "Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter," Proceedings of the NAACL Student Research Workshop, 2016, Published. https://doi.org/10.18653/v1/n16-2013

[2] J. H. Park, "One-step and Two-step Classification for Abusive Language," June 5, 2017,ArXiv.Org. https://arxiv.org/abs/1706.01206

[3] G. K. Pitsilis, “Detecting Offensive Language in Tweets Using Deep Learning,” January 132018, ArXiv.Org. https://arxiv.org/abs/1801.04433.

[4] S. MacAvaney, H. R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder,“Hate speech detection: Challengesandsolutions”, 2019,https://doi.org/10.1371/journal.pone.02211 52

[5] Z.Waseem, “Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter,” Proceedings of the First Workshop on NLP and Computational Social Science, Association for Computational Linguistics, 2016,doi:10.18653/v1/w16-5618.

[6] O.De Gibert, N.Perez, A.García-Pablos, and M.Cuadros, “Hate Speech Dataset from a White Supremacy Forum,” Proceedings of the 2nd Workshop on Abusive Language Online (ALW2), Association for Computational Linguistics, 2018,doi:10.18653/v1/w18-5102.

[7] R.Kumar, A. K.Ojha, S.Malmasi, and M.Zampieri, “Benchmarking Aggression Identification in Social Media,” Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018), Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp. 1–11, 2018,https://aclanthology.org/W18-4401.

[8] T.Mandl, S.Modha, P.Majumder, D.Patel, M.Dave, C.Mandlia, and A.Patel, “Overview of the HASOC track at FIRE 2019,” Proceedings of the 11th Forum for Information Retrieval Evaluation, ACM, 2019,doi:10.1145/3368567. 3368584.

[9] G.Kovács, P.Alonso,and R.Saini,“Challenges of Hate Speech Detection in Social Media,” SN Computer Science, vol. 2, no. 2, 2021, https://doi.org/10.1007/s42979-021-00457-3.

[10] M. Rizoiu, “Transfer Learning for Hate Speech Detection in Social Media,” June 10, 2019,ArXiv.Org. https://arxiv.org/abs/1906.03829

[11] A. Prabhu, “Towards Sub-Word Level Compositions for Sentiment Analysis of. . .,” November 2, 2016,ArXiv.Org. https://arxiv.org/abs/1611.00472

[12] Moshika, A., Thirumaran, M., Natarajan, B., Andal, K., Sambasivam, G., &Manoharan, R. (2021). Vulnerability assessment in heterogeneous web environment using probabilistic arithmetic automata. IEEE Access, 9, 74659-74673.

[13] P.Mathur, R.Sawhney, M.Ayyar,and R. Shah, “Did you offend me? Classification of Offensive Tweets in Hinglish Language,” Proceedings of the 2nd Workshop on Abusive Language Online (ALW2), 2018, Published. https://doi.org/10.18653/v1/w18-5118.

[14] A. Bohra, and D.Vijay, et al.,“A Dataset of Hindi-English Code Mixed Social Media Text for Hate Speech Detection,” ACLAnthology, June(2018, June,https://aclanthology.org/W18-1105/

[15] S. Kamble,and A. Joshi, “HateSpeechDetectionfromCode-mixed Hindi-EnglishTweets using Deep Learning Models,” November 13, 2018,ArXiv.Org. https://arxiv.org/abs/1811.05145

[16] T. Y.Santosh,and K. V. Aravind, “Hate Speech Detection in Hindi-English Code-Mixed Social Media Text,” Proceedings of the ACM India Joint International Conference on Data Science and Management of Data, 2019, Published. https://doi.org/10.1145/3297001.3297048.

[17] Rajesh, M., &Sitharthan, R. (2022). Image fusion and enhancement based on energy of the pixel using Deep Convolutional Neural Network. Multimedia Tools and Applications, 81(1), 873-885.

[18] N. Vashistha,and A.Zubiaga, “Online Multilingual Hate Speech Detection: Experimenting with Hindi and English Social Media,” Information, vol. 12, no. 1, p. 5, 2020.

[19] S.Gaikwad, T.Ranasinghe, M.Zampieri, and C. M.Homan, “Cross-lingual Offensive Language Identification for Low Resource Languages: The Case of Marathi,” 2021.

[20] S.Sazzed, “Identifying vulgarity in Bengali social media textual content,” PeerJ Computer Science, vol. 7, p. e665, 2021.

[21] A.Sharma, A.Kabra, and M.Jain, “Ceasing hate with MoH: Hate Speech Detection in Hindi–English code-switched language,” Information Processing andManagement, vol. 59, no. 1, p. 102760, 2022.

[22] S.Khanuja, D.Bansal, S.Mehtani, S.Khosla, A.Dey, B.Gopalan, D. K.Margam, P.Aggarwal, R. T.Nagipogu, S.Dave, S.Gupta, S. C. B.Gali, V.Subramanian, and P.Talukdar, “MuRIL: Multilingual Representations for Indian Languages,” 2021,arXiv preprint arXiv:2103.10730.

[23] Bird, Steven, Edward Loper and Ewan Klein,“Natural Language Processing with Python,” O'Reilly Media Inc., 2009.

[24] J. Devlin, “BERT: Pre-training of Deep Bidirectional Transformers for Language,” October 11, 2018,arXiv.Org. https://arxiv.org/abs/1810.04805

[25] T. Wolf, L. Debut, V.Sanh, J.Chaumond, C.Delangue, A.Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. le Scao, S. Gugger, and A. Rush, “Transformers: State-of-the-Art Natural Language Processing,” Proceedings of the 2020 Conference on Empirical Methods in NaturalLanguage Processing: System Demonstrations. 2020, https://doi.org/10.18653/v1/2020.emnlp-de mos.