

**MANGALORE**  **UNIVERSITY**

Title of the project

**“MULTILINGUAL CODE-MIXED HATE SPEECH DETECTION”**

**BY**

**RACHANA K**

**P05AZ21S0023**

**Under the Guidance of**

**Dr. H.L. Shashirekha**

**Professor**

**Department of Computer Science**

**Mangalore University**

**Mangalagangothri-574199**

DEPARTMENT OF POST GRADUATE STUDIES AND RESEARCH IN COMPUTER SCIENCE **MANGALORE UNIVERSITY**

**MANGALAGANGOTHRI-574199**

**KARNATAKA, INDIA**

**DECLARATION**

This is to declare the project entitled **“MULTILINGUAL CODE-MIXED HATE SPEECH DETECTION”** is a bonafide work independently carried out by me, **RACHANA K**, student of 3rd semester M.Sc. (Computer Science), under the supervision and guidance of  **Dr. H. L. Shashirekha**, professor, Department of Computer Science, Mangalore university. This is submitted in partial fulfilment of the requirement for the award of M.Sc. (Computer Science). Further, it is declared that this project is the result of my own efforts and has not been submitted to any other university for the award of any other degree or Diploma.

Place: Mangalagangothri RACHANA K

Date:  **Reg No :** P05AZ21S0023

**TABLE OF CONTENTS**

| **SL. No** | **Contents** | **Pg. No** |
| --- | --- | --- |
| **1** | **Abstract** | **4** |
| **2** | **Introduction** | **4-5** |
| **3** | **Literature Survey** | **5-7** |
| **4** | **Methodology** | **8-12** |
| **5** | **Experiment and Result** | **13-15** |
| **6** | **Conclusion** | **15** |
| **7** | **References** | **15** |

**ABSTRACT :**

Social media sites like Twitter, Facebook, being user-friendly and a free source, provide opportunities to people to air their voice. People, irrespective of the age group, use these sites to share every moment of their life making these sites flooded with data. Apart from these commendable features, these sites have downsides as well. Due to the lack of restrictions set by these sites for its users to express their views as they like, anybody can make adverse and unrealistic comments in abusive language against anybody with an ulterior motive to tarnish one’s image and status in the society. Hate speech and offensive language recognition in social media platforms have been an active field of research over recent years. In non-native English spoken countries, social media texts are mostly in code mixed or script mixed/switched form. Hate speech detection is a challenging problem with most of the datasets available in only one language: English. In this paper, we conduct a large-scale analysis of multilingual code-mixed hate speech in 7 languages from different sources. The experimental results showed that 1 to 3-gram character TF-IDF features are better for the said task. The best performing models were multinomial naive bayes, logistic regression and support vector machine for the datasets respectively.

**INTRODUCTION :**

Social media is a vast online communication forum that enables the public to express themselves easily, at times, anonymously. While expressing their opinion is a right of humans that is cherished, inducing and spreading offensive content towards another social community is an abuse of this liberty. Therefore, social media forums and other means of online communication platforms have begun to play a larger role in hate and offensive crimes.

Many online social media forums such as Twitter, Facebook, Instagram, and YouTube consider hate speech and offensive content harmful and have the policy to remove such content. Due to societal concern and how widespread offensive content is becoming on the Internet, there is a strong motivation to detect hate speech and offensive content in social media forums. Hate speech defines the attacks against someone or group community, based on these attributes as race, gender, ethnicity, religion, sexual orientation, age, physical or mental disability, and others. Offensive content is a language that could seriously offend an individual or group based on their age, religious or political beliefs, marital or parental status, sexual orientation, physical features, national origin, or disability.

India is a multistate, multilingual nation. Each state has its own official spoken language and respective script. The transliterated Romanized text of the native language with English as the binding language is termed as Codemix Text. Many people use codemixed text to create their social media contents.

**2. LITERATURE SURVEY :**

**KBCNMUJAL@HASOC-Dravidian-CodeMix-FIRE2020: Using Machine Learning for Detection of Hate Speech and Offensive Code-Mixed Social Media text :**

This paper is proposed by Varsha Pathak et al. This paper describes the system submitted by team KBCNMUJAL for Task 2 of the shared task “Hate Speech and Offensive Content Identification in Indo-European Languages (HASOC)” at FIRE 2020. The datasets consist of twitter messages with two class labels “offensive” and “not offensive”. The best performing classification models developed for both languages are applied on test datasets. The model which gives the highest accuracy result on training dataset for Malayalam language, was experimented to predict the labels of respective Test dataset. They used different features: word n-gram, character n-gram, combined word, character n-grams and custom word embedding. Using the TF-IDF weights of word and character n-gram features they trained their machine learning classifiers. Custom word embedding was used to train a simple neural network model. Their model for Malayalam language obtained an F1 score of 0.77 and for Tamil language Model they got F1 score as 0.87.

**Offensive language identification in Dravidian code mixed social media text:**

This paper is proposed by Sunil Saumya et al. The paper identifies the hate content in Tanglish, Manglish and Malayalam script mixed in tweets and validates with the dataset provided in HASOC-Dravidian-CodeMix-FIRE2020 challenge. The dataset proposed in the challenge was collected from Twitter. The experimental results showed that 1 to 6-gram character TF-IDF features are better for the said task. The best performing models were naive bayes, logistic regression, and vanilla neural network for the dataset Tamil code-mixed, Malayalam code-mixed, and Malayalam script-mixed, respectively. They also examined a few transfer learning models like BERT and ULMFit for the classification task. For Tamil code-mixed, VNN reported precision, recall, and F1-score of 0.89 and for Malayalam codemixed it reported precision, recall, and F1-score of 0.77. Similarly, for Malayalam script-mixed data, the proposed vanilla neural network reported a precision, recall, and F1 score of 0.95.

**Detection of Hate Speech Text in Hindi-English Code-mixed Data:**

This paper is proposed by Sreelakshmi ka et al. This paper describes the machine learning model to detect hate speech in Hindi-English code-mixed social media text. The methodology makes use of Facebook’s pre-trained word embedding library, fastText to represent 10000 data samples collected from different sources as hate and non-hate. The performance of the proposed methodology is compared with word2vec and doc2vec features and it is observed that fastText features gave better feature representation with Support Vector Machine (SVM)-Radial Basis Function (RBF) classifiers. The paper also provides an insight to the researchers working in the field of code-mixed data that character level features provide best results for code-mixed data. FastText features gave the highest accuracy of 85.81% using SVM- RBF as classifier followed by word2vec with 75.11% accuracy using SVM-RBF and then doc2vec with an 64.15% accuracy using Random Forest.

**HASOCOne@FIRE-HASOC2020: Using BERT and Multilingual BERT models for Hate Speech Detection :**

This paper is proposed by Suman Dowlagara et al. In this paper they propose an approach to automatically classify hate speech and offensive content. They used the datasets obtained from FIRE 2019 and 2020 shared tasks. They perform experiments by taking advantage of transfer learning models. They observed that the pre-trained BERT model and the multilingual-BERT model gave the best results. They used pre-trained bi-directional encoder representations using transformers (BERT) and multilingual BERT for hate speech and offensive content detection for English, German, and Hindi languages. For the english dataset on hate speech and offensive content detection using BERT model they got 88.33% and 81.57% accuracy respectively. For the German dataset using multilingual-BERT they got 82.51% and 80.42% respectively. Also for the hindi dataset using multilingual-BERT they got 74.96% and 73.15% respectively.

**3. METHODOLOGY:**





















**3.1 PRE-PROCESSING**

**3.1.1 CONVERT EMOJI INTO TEXT :**

Emojis are replaced by the words that the emoji represents like happy,sad, among other emotions depicted by emojis. As emojis mainly depict a user’s intention, it would be imperative to replace them with their meanings to pick up their cues.

**3.1.2 REMOVING URL :**

URLs (or Uniform Resource Locators) in a text are references to a location on the web, but do not provide any additional information.

**3.1.3 EXPANDING CONTRACTED WORDS :**

A contraction is a shortened form of a group of words. Contractions are used in both written and oral communication, a lot of us tend to contract common words like “you are” becomes “you’re”. Converting contractions into their natural form will bring more insights.

**3.1.4 REMOVING PUNCTUATION AND DIGITS:**

The marks are used in writing to separate sentences and their elements, as well as to clarify meaning, such as the full stop, comma, and brackets. They aid stemming and have no meaning if they are removed. Also digits present in data are removed.

'!"#$%&\'()\*+,- ./:;<=>?@[\\]^\_`{|}~'

**3.1.5 REMOVING STOP WORDS :**

Stop words are a set of commonly used words in a language. Stop words are commonly used in Text Mining and NLP to eliminate words that are so commonly used that they carry very little useful information.

Some of the stopwords are :

{ಮತ್ತು,अंदर,এই,आहे,ஒரு,a,ಅವರು,ಎಂಬ,ಮೇಲೆ,आप,इत्यादि,इन,আমরা,প্রায়,দুই,நான்,

உள்ள,அந்த,கொள்ள,இந்தத்,also,although,always}

**3.2 FEATURE EXTRACTION**

Feature extraction is the process of extracting feature from the text and then transforming them into a feature set which is usable by a classifier. TF – IDF method is used for feature extraction. The TF-IDF originated for the documented analysis and retrieval of information from text.

**TERM FREQUENCY**

The higher a word appears in a document, the more relevant it is. This might be correct but what happens when the documents are of variable sizes, how is it supposed to determine which word is more relevant to all documents. Term frequency is defined as a number of times an instance or keyword appears in a single document divided by the total number of words in that document.

**INVERSE DOCUMENT FREQUENCY**

Inverse Document Frequency calculates the weight of a rare term of the text in a collection of documents.

We need to convert text data into vectors as a machine learning algorithm works only on numeric data. For this we will use TF-IDF.

**3.3 MACHINE LEARNING ALGORITHMS**

**3.3.1 SUPPORT VECTOR MACHINE(SVM)**

model = LinearSVC()

SVM is mapping data to a high-dimensional feature space so that data points can be categorised, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane.

The objective of the SVM algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. The dimension of the hyperplane depends upon the number of features.

model.fit(X\_train, Y\_train)

**3.3.2 LOGISTIC REGRESSION**

model = LogisticRegression()

Logistic regression is used to predict the probability of certain classes based on some dependent variables. The output of logistic regression is always between (0, and 1), which is suitable for a binary classification task. The higher the value, the higher the probability that the current sample is classified as class=1, and vice versa. The output is the prediction value when the value is closer to 1, which means the instance is more likely to be a positive sample(y=1). If the value is closer to 0, this means the instance is more likely to be a negative sample(y=0).

Training the logistic regression model with the training data

model.fit(X\_train, Y\_train)

**3.3.3 MULTINOMIAL NAIVE BAYES**

model = MultinomialNB()

Multinomial Naive Bayes is a probabilistic model and specialised version of Naive Bayes. Simple Naive Bayes model a document as the presence and absence of particular words, whereas Multinomial Naive Bayes explicitly models the word counts and adjusts the underlying calculations to deal with it. Multinomial Naive Bayes is one of the variations of the Naive Bayes algorithm in machine learning which is very useful to use on a dataset that is distributed multinomial. When there are multiple classes to classify, this algorithm can be used because to predict the label of the text it calculates the probability of each label for the input text and then generates the label with the highest probability as output.

Training the Naive Bayes model with the training data

model.fit(xtrain\_tfidf, y\_train)

**4. EXPERIMENT AND RESULT**

**4.1 DATASET**

The dataset used in this experiment is taken from Github, Hasoc and Kaggle. The dataset consists of 7 languages such as Bengali, Kannada, English, Hindi, Tamil, Malayalam, Marathi and the total size of the dataset is 145371.

In this, the size of the training set and test set is 104559 and 34854 respectively.

| **Languages** | **Labels** | | **Total** |
| --- | --- | --- | --- |
| **Hate** | **Not\_hate** |
| **Kannada** | **1311** | **3970** | **5281** |
| **Malayalam** | **627** | **15332** | **15959** |
| **Tamil** | **9283** | **28618** | **37901** |
| **Marati** | **1329** | **2284** | **3613** |
| **Hindi** | **6393** | **10428** | **16821** |
| **Bengali** | **10000** | **20000** | **30000** |
| **English** | **26096** | **9080** | **35176** |

**4.2 EXPERIMENTS**

We collected datasets from various sources but it contains several duplicate data. Since we removed duplicate data from the original dataset and further we used that formatted data for the process. TF – IDF vectorizer is used for feature extraction. Then machine learning algorithms such as logistic regression(LR), multinomial naive bayes and Support Vector Machine(SVM) are used .

**4.3 RESULT**

Class-wise Precision ,Recall and F1-Scores for both the classes of the dataset

| **MODEL** | **HATE** | | | **NOT\_HATE** | | | **ACCURACY** |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Precision** | **Recall** | **F1 Score** | **Precision** | **Recall** | **F1 Score** |  |
| **SVM** | **0.88** | **0.71** | **0.79** | **0.83** | **0.94** | **0.88** | **0.85** |
| **LR** | **0.87** | **0.71** | **0.78** | **0.83** | **0.93** | **0.88** | **0.85** |
| **MNB** | **0.84** | **0.68** | **0.75** | **0.81** | **0.91** | **0.86** | **0.82** |

The experimental results of classifying multilingual code-mixed hate speech with various distinct techniques are shown in terms of precision and recall for the respective classes, along with the overall accuracy being tabulated in Table.

The three metrics are computed as follows:

Precision = T P / (TP + FP)

Recall = T P/ ( T P + F N)

F1 − Score = (2 ∗ P ∗ R)/( P + R)

TP, FP, and FN are True Positives, False Positives, and False Negatives.

**5. CONCLUSION**

Hate speech detection is a challenging problem with most of the datasets available in only one language i.e english. We used 7 languages collected from different sources and used the TF-IDF method for feature extraction. We trained data using different machine learning models such as logistic regression,support vector machine and multinomial naive bayes. Out of these SVM algorithms gives the best accuracy obtained as 85%.

**6. REFERENCES**

1. Varsha Pathak et al KBCNMUJAL@HASOC-Dravidian-CodeMix-FIRE2020:

Using Machine Learning for Detection of Hate Speech and Offensive Code-Mixed Social Media text.

1. Sunil Saumya et al. : Offensive language identification in Dravidian code mixed social media text
2. Sreelakshmi ka et al. : Detection of Hate Speech Text in Hindi-English Code-mixed Data
3. Suman Dowlagara et al. : HASOCOne@FIRE-HASOC2020: Using BERT and Multilingual BERT models for Hate Speech Detection