International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 04 | April 2022 www.irjet.net p-ISSN: 2395-0072

Language and Offensive Word Detection

Akash Kotekar[1], Anuj Jaijeevan[2], Tauqeer Rumaney[3], Vishakh GR[4], Prof. Shubhangi Chavan[5]

[1,2,3,4] *UG Student, Dept. of Computer Engineering, Pillai College of Engineering, New Panvel, India.*

[5] *Assistant Professor, Dept. of Information Technology, Pillai College of Engineering, New Panvel, India.*

Abstract— *Language recognition is a task in natural language processing that recognizes the natural language that makes up a document's content automatically. Language recognition helps users and machine translation communicate more effectively. In many NLP applications, language recognition is a fundamental and crucial stage. Machine learning and n-gram-based language identifiers are used to train and recognize numerous languages in this research. The first and most important step in using a code mixed language translation tool is to identify the language. It's also found in tools for multilingual summarizing and paraphrasing.*

Our software is also aimed at detecting offensive terms in any type of text provided by the user. People that engage in some type of online content (for example, articles) targeting an individual or a group have grown in popularity as social media has grown in prominence in recent years. Our programme will analyze the text using natural language processing or machine learning methods and compare it to the dataset to see if it contains any offending terms.

Keywords—Language recognition, multilingual, offensive word, NBSVM.

1. Introduction

Language identification is a task in natural language processing that recognizes the natural language in which the content of a document is written automatically. It is vital to determine the language of the content before employing any natural language application. Language identification is a key and crucial stage in many NLP applications.

Our technology also aims to detect offensive terms in any sort of text that the user provides. A text is considered threatening or abusive if it contains sexist or racist slurs, targets or condemns any community or religious perspective, or stimulates criminal conduct. Our programme will compare the text to the dataset and analyze it using natural language processing or machine learning methods to check whether it contains any offensive terms.

2. Literature Survey

A. Language Detection Engine for Multilingual Texting on Mobile Devices: Sourabh Vasant Gothe, Sourav Ghosh, Sharmila Mani, Bhanodai Guggilla, Ankur Agarwal, and Chandramouli Sanchi introduced the Language Detection Engine (LDE), a system that improves multilingual typing user experience by precisely determining the language of input text in real-time. LDE is a combination of a character N-gram model, which calculates the chance of input text coming from a specific language, and a selector model, which employs the emission probabilities to determine the most likely language for a given text using logistic regression.

B. Automatic Hate Speech Detection on Social Media: A Brief Survey: Ahlam Alrehili [3] proposes a comprehensive and state-of-the-art natural language processing (NLP) technique for automatic hate speech identification on OSNs in this study. We focused on eight frequently used strategies for automatic hate detection, with N-gram being the most efficient and user-friendly.

C. Language Identification for Multilingual Machine Translation: S Arun Babhulgaonkar and Shefali Sonavane [1] developed n-gram and machine learning-based language identifiers and used them to identify three Indian languages in a document submitted for machine translation: Hindi, Marathi, and Sanskrit. Incorporating language identification components into machine translation enhanced translation quality.

2.1 Summary of Related Work

The summary of methods used in literature is given in Table 1.

Table 1 Summary of literature survey

Paper	Advantages	Disadvantages
S Arun Babhulgaonkar et al. 2020 [1]	This paper shows that SVM gives better results than other Language identifiers. .	It is also observed that most of the misclassified instances are short and noisy
Sourabh Vasant Gothe et al. 2020 [2]	LDE accurately detects codeswitching in a multilingual text with the help of a uniquely designed selector model.	It is a shallow learning model..
Ahlam Alrehili 2019 [3]	This paper gives a detailed study of the commonly used hate speech detection techniques.	Only eight technique comparisons are done.
P. Mathur et al. 2018 [4]	The success of transfer learning for analyzing complex cross linguistic textual structures can be extended to include many more tasks involving code-switched and code-mixed data.	Hinglish tweets in the dataset suffer from syntactic degradation after transliteration and translation which leads to a loss in the contextual structuring of the tweets.

Below is overview of comparison of different parameters

Table 2 Summary of literature survey

Types of Classifier	Language	Accuracy
n-gram based Language Identifier	Hindi	73.58%
n-gram based Language Identifier	Marathi	76.53%
n-gram based Language Identifier	Sanskrit	76.04%
Logistic Regression	Hindi	81.81%
Logistic Regression	Marathi	79.80%
Logistic Regression	Sanskrit	81.44%
Support Vector Machine	Hindi	87.68%
Support Vector Machine	Marathi	90%
Support Vector Machine	Sanskrit	89.23%
Naïve Bayes Classifier	Hindi	75.96%
Naïve Bayes Classifier	Marathi	76.99%
Naïve Bayes Classifier	Sanskrit	79.16%

3. Proposed Work

Our research is aimed at recognizing 44 distinct languages and offensive terms in any type of text that the user provides as input. Because social media has grown in popularity in recent years, there are people who engage in some type of online material (for example, articles) aimed towards a person or a group, etc. So, using natural language processing or machine learning methods, our programme will evaluate the text and refer back to the dataset to determine the language of the input text as well as whether or not it contains any offending terms (for English and hinglish only).

3.1 System Architecture

The proposed system architecture is given in Figure 1.

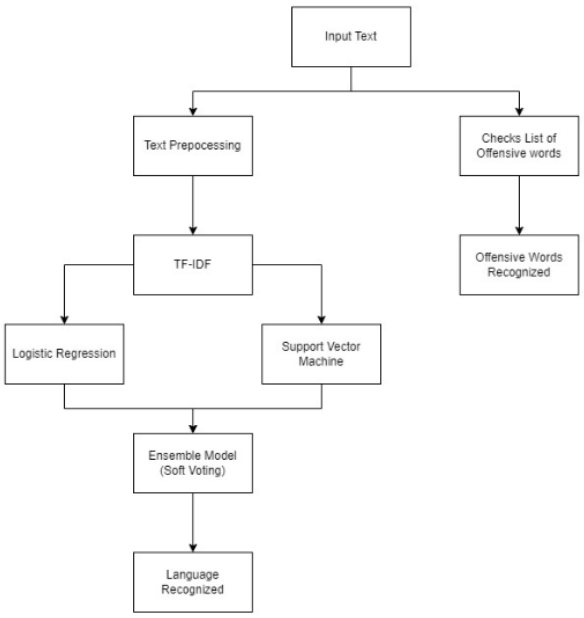


Fig. 1 Proposed system architecture

A. Dataset Creation:

We have collected datasets for 44 different languages and cleaned them by performing filtration on it. Then all the datasets were combined into one single dataset containing 69290 sentences.

For offensive word detection, we downloaded two (english and hinglish) datasets, cleaned and merged them into one dataset containing 1712 words.

B. Training and Testing:

The dataset for language detection was split into a training (80%)(48503 sentences) and testing (20%)(20787 sentences) set. This dataset was trained and tested using the Logistic Regression and Tf-Idf model.

C. TF-IDF Model:

The TF-IDF (term frequency-inverse document frequency) statistic examines the relevance of a word to a document in a collection of documents. This is accomplished by multiplying two metrics: the number of times a word appears in a document and the word's inverse document frequency over a collection of documents.

D. Logistic regression Model

Logistic regression is a statistical machine learning technique that classifies data by taking extreme outcome variables and attempting to draw a logarithmic line that separates them.

E. Support Vector Machine Model

The "Support Vector Machine" (SVM) is a supervised machine learning technique that can solve classification and regression issues. Each data item is plotted as a point in n-dimensional space (where n is the number of features you have), with the value of each feature being the value of a certain coordinate in the SVM algorithm. Then we accomplish classification by locating the hyper-plane that clearly distinguishes the two classes.

F. Ensemble Model (Soft Voting)

Ensemble learning is the process of systematically generating and combining many models, such as classifiers or experts, to tackle a specific computational intelligence problem.

Models that forecast class membership probability are subject to soft voting. Soft voting can be utilized for models that don't predict class membership probability natively, but it may require some calibration of their probability-like scores before they can be employed in the ensemble (e.g. support vector machine, k-nearest neighbors, and decision trees). When you have two or more models that perform well on a predictive modelling job, you should use a voting ensemble. The ensemble models must agree on the majority of their forecasts.

E. N-gram (Tokenization)

In a document, N-grams are continuous sequences of words, symbols, or tokens. N-gram, in our model, decomposes a phrase into a series of tokens containing each word. We use N-gram as a tokenizer to identify

offensive phrases where two or more words combine to generate an offensive phrase.

D. Flow of Project

Language Detection –

1) Input Text
2) Preprocessing -
    a) Text converted to Lowercase
    b) Removing Numbers
    c) Removing Punctuations
3) Feature Extraction -
    a) Tf-idf of the clean input text
4) Model -
    a) Feeding feature matrix into Soft Voting classifier(Logistic Regression and Support Vector Machine)
5) Language Recognized

Offensive Word Detection –

1) Input Text
2) Preprocessing -
    a) Text converted to Lowercase
    b) Removing Numbers
    c) Removing Punctuations
3) Feature Extraction -
    a) Tokenization using N-gram
4) Comparing tokens with the list of offensive words in the dataset.
5) Offensive Words Recognized

3 Requirement Analyses

The implementation detail is given in this section.

3.1 Software

Table 3.3 Software details

Operating System	Windows 10
Programming Language	Python

3.2 Hardware

Table 3.2 Hardware details

Processor	2 GHz Intel
HDD	180 GB
RAM	2 GB

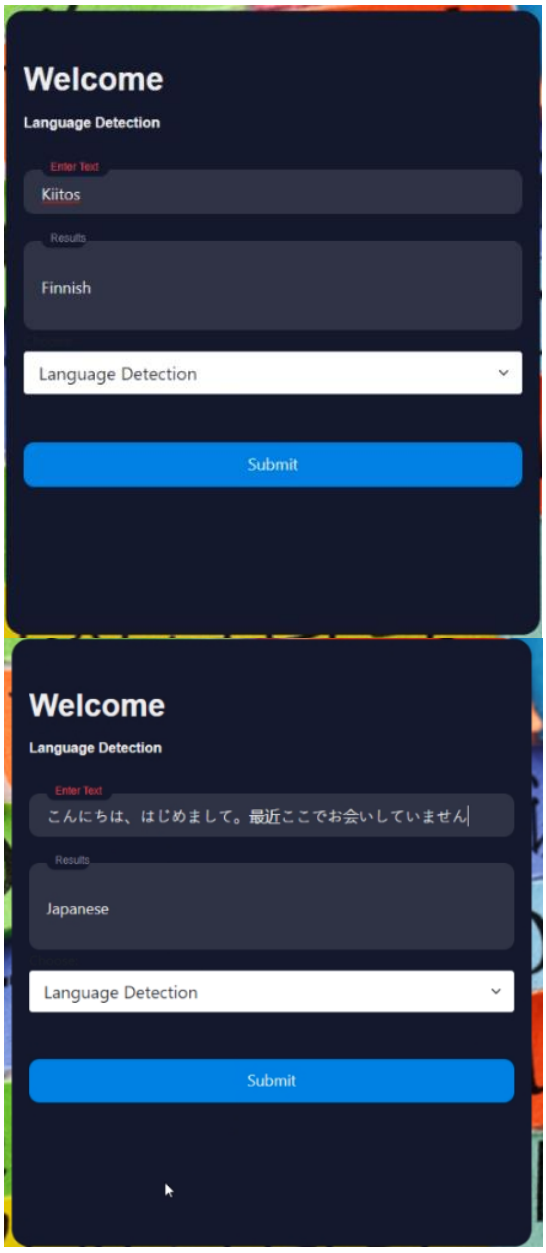
3.3 Dataset

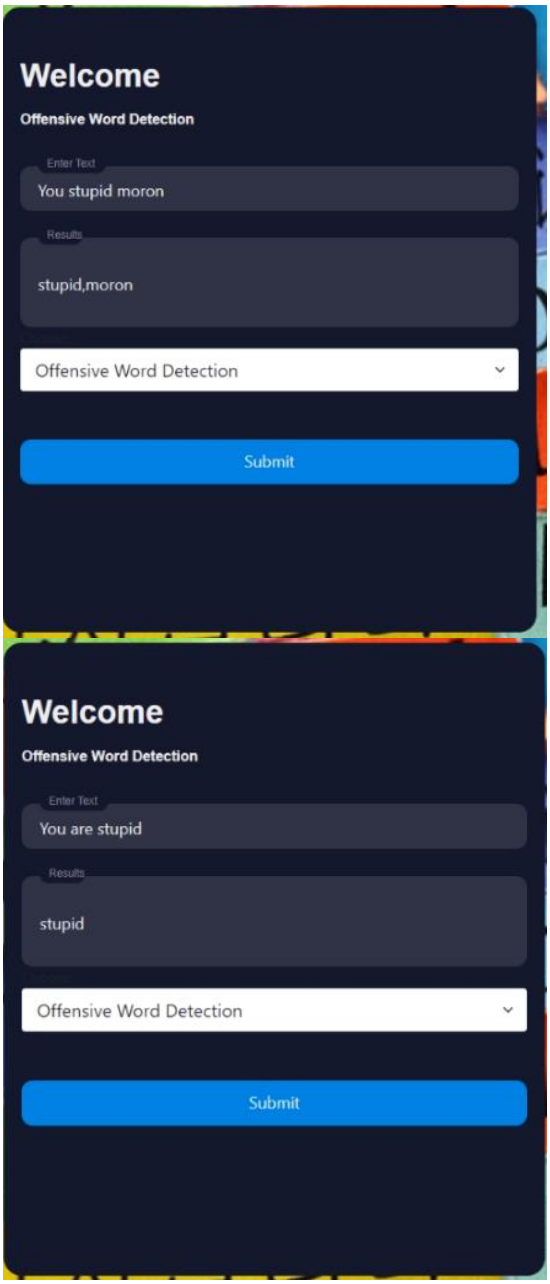
The Language Detection dataset, which provides text details for several languages, is used. We must construct a model that will be able to predict the given language using the text. We begin by loading the dataset and performing some preliminary processing. We start by filtering the data to find statements of the right length and language. These sentences are then divided into three groups: training (70%), validation (20%), and test (10%). We need to extract features from our collection of phrases to generate a feature matrix before we can fit a model.

4 Conclusions

The SVM (Support Vector Machine) classifier had the best accuracy among the other classifiers, but we chose the Logistic Regression model after attaining a 99.2 percent accuracy from training and testing the dataset.

We employed a simple method for recognizing offensive words for offensive word identification, but we also used N-gram for a more advanced search in the dataset.





Acknowledgment

It is our privilege to express our sincerest regards to our supervisor Shubhangi Chavan for the valuable inputs, able guidance, encouragement, whole-hearted cooperation and constructive criticism throughout the duration of this work. We deeply express our sincere thanks to our Head of the Department Dr. Sharvari Govilkar and our Principal Dr. Sandeep M. Joshi for encouraging and allowing us to present this work.

REFERENCES

1. S Arun Babhulgaonkar, Shefali Sonavane; "Language Identification for Multilingual Machine Translation", 2020 International Conference on Communication and Signal Processing (ICCSP).
2. Sourabh Vasant Gothe, Sourav Ghosh, Sharmila Mani, Bhanodai Guggilla, Ankur Agarwal, Chandramouli Sanchi; " Language Detection Engine for Multilingual Texting on Mobile Devices", 2020 IEEE 14th International Conference on Semantic Computing (ICSC).
3. Ahlam Alrehili, "Automatic Hate Speech Detection on Social Media: A Brief Survey", in: IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), Nov. 2019.
4. P. Mathur, R. Shah, R.Sawhney, and D. Mahata, "Detecting offensive tweets in hindi-english code-switched language," in Proceedings of the Sixth International Workshop on Natural Language Processing for SocialMedia,2018.
5. A. H. Razavi, D. Inkpen, S. Uritsky, S. Matwin; "Offensive Language Detection Using Multi-level Classification", Canadian Conference on AI 2010.
6. Puja Chakraborty, Md. Hanif Seddiqui; "Threat and Abusive Language Detection on Social Media in the Bengali Language", 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT).
7. Alina Maria Ciobanu, Marcos Zampieri, Shervin Malmasi Santanu Pal, Liviu P. Dinu; "Discriminating between Indo-Aryan Languages Using SVM Ensembles", 2018 "In Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects" Association for Computational Linguistics, USA.
8. Gabriel Araújo De Souza, Márjory Da Costa-Abreu, "Automatic offensive language detection from Twitter data using machine learning and feature selection of metadata" in International Joint Conference on Neural Networks (IJCNN), 19-24 July 2020.