TENGLISH ABUSIVE COMMENT DETECTOR

Naga Durga Saile. K, Assistant Professor, Department of CSE-AIML & IoT, VNR VJIET.
Nalla Sai Krishna Reddy, Jami Venkata Sai, Yashwanth Shankar Gande, Student, Department of CSE-AIML & IoT, VNR VJIET.

Abstract

The rise of online communication has led to an increased occurrence of abusive language and offensive content in various languages like English, Tamil, and Telugu etc. The need for effective tools to identify and remove such kind of content has become crucial to maintain a healthy online environment. This research includes a Telugu Abusive Comment Detection System using Machine Learning techniques. The proposed system uses a dataset of manually labelled Telugu comments including abusive and non-abusive language. The ML model is trained on features from the text. Various algorithms, such as Support Vector Machines (SVM), Naive Bayes, and deep learning models, are used to detect abusive language in Telugu. To enhance the system's performance, a pre-processing pipeline is implemented to handle these comments in Telugu. Tokenization, stemming, and stop-word removal are used to extract meaningful features from the text. Additionally, the model is fine-tuned using an evaluation framework to ensure its effectiveness in real-world scenarios. The Telugu Abusive Comment Detector aims to provide a valuable tool for social media platforms, online forums, and other digital spaces where abusive language poses a threat to user experience, community well-being and a healthy environment. Successful implementation of this system contributes to developing a safer online environment for Telugu-speaking users.

Keywords: Abusive Language, Telugu-English Language, Pre-processing, Machine Learning Models

1. Introduction

In this rapidly evolving landscape of online communication, user-generated content has given rise to an alarming increase in the usage of abusive language across various languages like English, Telugu, Tamil, Malayalam and many more. One such language facing the challenges of online abuse is Telugu, a language spoken predominantly in the Indian states of Andhra Pradesh and Telangana. The need for robust tools to detect and remove abusive comments in Telugu has become necessary to ensure the creation of a safer online environment for its users. This research is to address this critical issue through the development of a sophisticated Telugu Abusive Comment Detection System by using the power of various Machine Learning (ML) techniques. The system is designed to analyse and categorize Telugu language comments into abusive and non-abusive categories, by using a dataset that contains a range of linguistic expressions commonly found in online interactions, i.e. on several online platforms such as Instagram, YouTube, Discord, Snapchat, Reddit etc. The primary focus of this system lies in the utilization of state-of-the-art ML algorithms, including Support Vector Machines (SVM), Naive Bayes, and several other models like XG Boost Regressor etc. These algorithms are trained on features obtained from the Telugu text to enable accurate detection of abusive language patterns. Recognizing the patterns of Telugu linguistics, this system uses a deep pre-processing pipeline to process the comments. Tokenization, stemming, and stop-word removal are implemented to ensure the extraction of features from the Telugu comments, enhancing the overall efficiency of the ML model. Furthermore, the system undergoes rigorous evaluation processes framework to increase its reliability and effectiveness in real-world scenarios. The objective is to create a solution that not only demonstrates perfection in detecting abusive language but also to ensure a low rate of false results, i.e. false positives, and false negatives, hence promoting user trust and confidence in the model's capabilities. The proposed Telugu Abusive Comment Detection System emerges as a crucial tool for online platforms, social media networks, and digital communities where the issue of abusive language threatens user experience and community’s well-being. By successfully implementing this ML solution, the research contributes significantly to the ongoing efforts to bring a safer and more respectful online environment for Telugu-speaking users. The following part of the document is organised as Related work, Materials used, Machine Learning Model Training and the last section describes the Conclusion and Future Work

2. Literature survey

The authors in [1] present research on detecting abusive comments in Tamil and Tamil-English code-mixed text from YouTube comments. Two datasets were created - one for Tamil comments and one for Tamil-English code-mixed comments. The datasets contain YouTube comments which were annotated for abusive speech at two levels - binary (abusive vs non-abusive) and fine-grained (misogyny, homophobia, transphobia, xenophobia, counter speech, hope speech, none). Machine learning models like SVM, RF, LR, NB, DT and deep learning models like LSTM, CNN were experimented with using different feature representations like TF-IDF, Bag-of-Words, Fast Text. For binary abusive comment detection, the multilingual transformer model MURIL performed the best, indicating suitability of multilingual models for this task. For fine-grained detection, traditional ML models outperformed deep learning models due to fewer samples per class. TF-IDF and BoW features worked better than Fast Text. The work demonstrates the need for a fine-grained approach to abusive speech detection in low-resource languages like Tamil. The datasets and experiments establish baselines for future work.

This paper [2] describes the Abusive language detection from social media comments using conventional machine learning and deep learning approaches. The authors worked on abusive language detection from Urdu and Roman Urdu comments using five diverse ML models (NB, SVM, IBK, Logistic, and JRip) and four DL models (CNN, LSTM, BLSTM, and CLSTM). The experiments, the authors found that that the convolutional neural network outperforms the other models and achieves 96.2% and 91.4% accuracy on Urdu and Roman Urdu.

The authors of [3] describes the Comment Abuse Classification with Deep Learning techniques of a recurrent neural network (RNN) with a long-short term memory cell (LSTM) and word embeddings, a convolutional neural network (CNN) with word embeddings, and a CNN with character embeddings. The models improve upon previous results from non-deep learning machine-learning models, and it is observed that a CNN with character-level embeddings reaches the highest performance.

[4] Presents the techniques for Abusive language Detection using Machine Learning where the data is gathered from the websites It has been shown that most participating users who provide comments during a certain occurrence are likely to degrade the victim.

3. Material

The dataset used for training the model is obtained through various sources, i.e. through forms, web-scraping and they have been manually labelled so that the training process is more effective and the results are more reliable. Google forms have been used as a method to add data to the dataset, i.e. asking people to fill in abusive comments or words they have seen in the past so that the model can be trained to find those abusive comments. Majority of the data in the dataset has been obtained from Web-Scraping. Web scraping is the technique used to extract data from websites. It involves fetching the HTML code of a web page and extracting data from it. Using Web-Scraping we have scraped several comments off Instagram reels which mainly consisted Telugu reels to train the model on Telugu dataset. Instagram is a very widely used platform and extracting comments from reels also contributes a lot to the dataset. The total Size of the dataset is 6K which are collected from various sources. Methods Used for collection of the comments is in the following process which are discussed as the process of data extraction using three different methods.

3.1 Extraction of Data

3.1.1 Type 1 (When reels section is opened directly):

1. Catch comment fetching network call in Instagram’s network calls.

2. Copy the content of the JSON object received.
3. Paste this object in the file below.
4. All the comments will be extracted into data.txt file.

3.1.2 Type 2 (When reels are opened from message):

1. Catch comment fetching network Call in Instagram's network calls.
2. Copy the content of the JSON object received.
3. Paste this object in the file below.
4. All the comments will be extracted into data.txt file.

3.1.3 Type 3 (using IG Comment Exporter Extension in Chrome):

1. Paste Instagram post's URL and comments are extracted in a csv or excel file.

3.2 Final Steps

1. Comments obtained in data.txt file through type 1,2 methods are pasted into master csv file.
2. Comments obtained in csv file through type 3 method are pasted into master csv file
3. Manually all comments are labelled as either abusive or non-abusive for model training

4. Machine Learning Model Training

The progression of this study initially involved a dataset comprising a modest 200 data points, presenting an initial accuracy plateau of around 72% when subjected to various machine learning models. This served as the foundational bedrock for subsequent explorations. To enhance the model's performance and robustness, the dataset was substantially expanded to 3000 data points, resulting in a noteworthy surge in accuracy to approximately 82%.

Motivated by the encouraging results observed through this augmentation, a further step was taken to enrich the dataset, scaling it up to a more comprehensive compilation of 6219 data points. This amplified dataset was then meticulously fed into an array of diverse machine learning algorithms, each designed to extract distinct patterns and relationships inherent in the data. These algorithms encompassed an array of classifiers, including Linear Kernel SVC, Gaussian RBF Kernel SVC, Decision Tree Classifier operating with the criterion set to 'Log Loss' and utilizing the 'best' splitter, KNN Classifier with a neighbour count of 5 and 'minkowski' as the metric, Logistic Regression, Gaussian Naïve Bayes, Random Forest Classifier incorporating 20 estimators, and Boost Classifier. An innovative stacking approach was adopted, wherein the outputs of these individual models were amalgamated using a Stacking Classifier technique. This stacking methodology culminated in a final estimator,

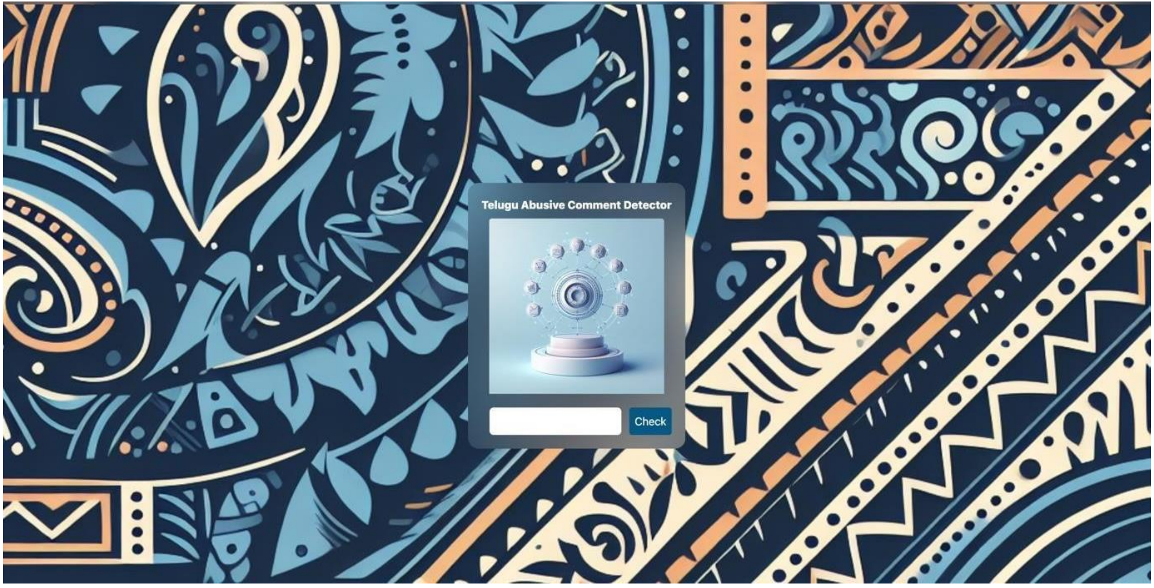
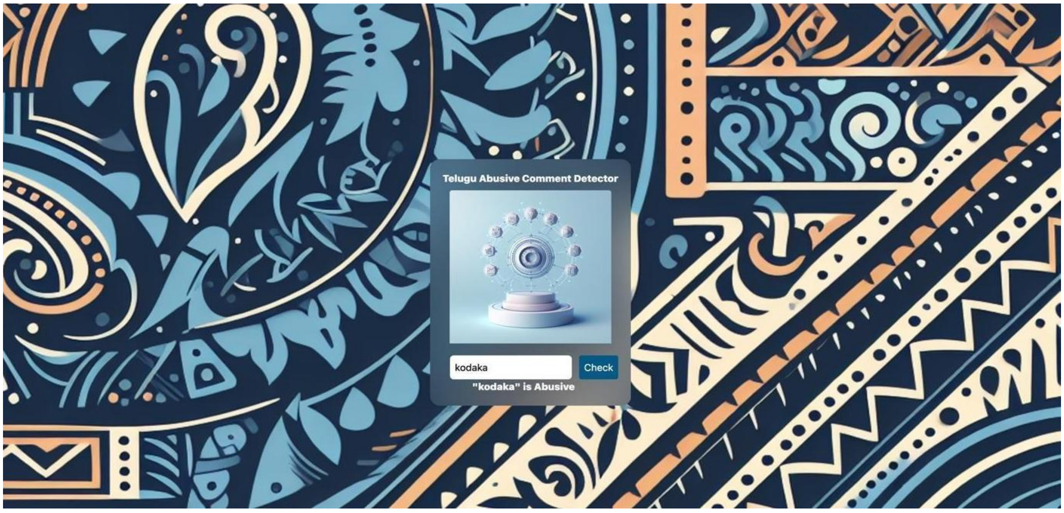


Fig 4.1 Enter the Comment

deploying the reliable Linear Kernel SVC. The amalgamation of these diverse models led to a substantial elevation in accuracy, achieving an impressive 87.78% with a measured standard deviation of 1.12%. The consequential increase in accuracy, coupled with the relatively low standard deviation, signified a robust and promising predictive capacity of the unified model.

As a critical step towards practical deployment and broader utilization, the meticulously engineered and validated model was meticulously exported utilizing the Pickle library. This

transformation facilitated the seamless integration of the model into a web interface, paving the way for practical real-world applications and facilitating user interaction with the proficient predictive capabilities encapsulated within the model.



kodaka

"kodaka" is Abusive

Fig 4.2 The model predicts the comment as Abusive

5. Experiment Findings and Analysis

5.1 Data Preprocessing

The raw data included text in Telugu as well as emojis. Preprocessing steps were taken to clean the data before vectorization. First, emojis were removed using regular expressions in Visual Studio Code. Second, comments containing only Telugu text without any English letters a-z or A-Z were deleted. English words in the remaining comments were reduced to their root form to decrease the number of columns after vectorization. The pre-processed comments were then converted into numerical feature vectors using TF-IDF (Term Frequency - Inverse Document Frequency) vectorizer. This vectorized dataset was used to train the model further.

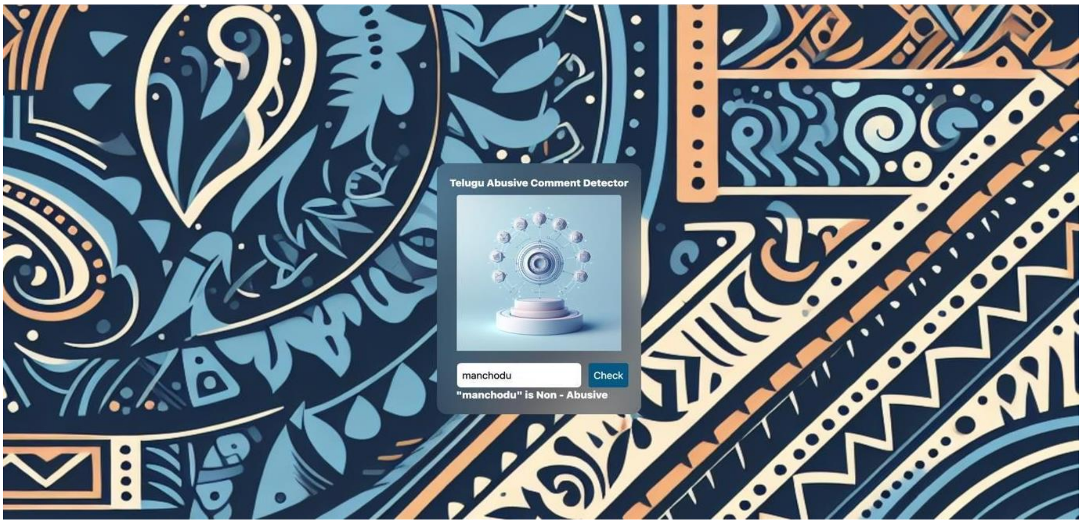
5.2 Strategy

The dataset comprises of 6000 comments in which few are web scraped and few are collected through the online web form. The data is split into training and testing data in the ratio 8:2 i.e. 80% training data and 20% testing data.

5.3 Results

S. No	Algorithm	Accuracy
1	Decision Tree Classifier	86.33%
2	**Stacking with Linear SVM as FC**	**89.79%**
3	Linear SVM	88.10%
4	Logistic Regression	85.93%
5	Naive Bayes	63.98%
6	Random Forest Classifier	88.58%
7	RBF SVM Classifier	86.97%
8	XGBoost Classifier	87.62%
9	K-Nearest Neighbours	79.90%

Table1: Results with Various Classifiers



Telugu Abusive Comment Detector

manchodu

"manchodu" is Non - Abusive

Fig. 3 The model predicts the comment as non-abusive

6. Conclusion and Future Scope

In this work, we presented a comprehensive machine learning framework for detecting abusive comments in Telugu language. Automated abusive content moderation can promote healthier conversations online and address issues like cyberbullying. A labelled dataset of Telugu social media comments was utilized along with text pre-processing and vectorization. Various traditional classifiers were evaluated including Support Vector Machines, Decision Trees, Logistic Regression and XGBoost.

However single models tend to overfit on textual data. To improve robustness, we proposed an ensemble by stacking multiple base classifiers. The outputs of diverse base models were combined using a meta-classifier. This enhanced overall accuracy since noise from an individual model gets suppressed when predictions are aggregated. Our stacking ensemble model achieved over 87.78% accuracy in identifying abusive comments using stratified cross-validation. We also analysed precision, recall and K-fold cross validation score for comprehensive evaluation.

The results demonstrate significant improvement over single classification algorithms, highlighting the efficacy of stacked ensembles. In future, larger labelled Telugu datasets can further boost performance. Advanced deep learning approaches using recurrent neural networks and attention mechanisms also hold promise. From an application perspective, the abusive comment detection framework can be potentially integrated with comment systems, social sites, and forums in Telugu to automatically moderate inappropriate content. Overall, this work makes valuable contributions in an important yet relatively less explored area of abusive text detection specifically for Telugu language on social media. The techniques proposed can provide guidelines for similar text classification tasks involving other regional languages as well. Automated moderation combined with human oversight can lead social media platforms toward more constructive public discourse. The future scope is to expand to other regional Indian languages like Hindi, Tamil, Kannada by training language- specific models. To Include social context like author profiles, network interactions to understand the intent behind abusive posts. To Add multimodal capabilities for identifying abusive images, audios, videos based on ML. Enable user-level abusive behaviour analysis by identifying problematic accounts. Build automated moderation tools like comments lockdown powered by the detection model. Integrate the model via site APIs for instant flagging and removal of abusive content.

References

[1] Bharathi Raja Chakravarthi, Ruba Priyadharshini, Shubanker Banerjee, Manoj Balaji Jagadeeshan, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Sean Benhur, John Philip McCrae,

Detecting abusive comments at a fine-grained level in a low-resource language, Natural Language ProcessingJournal,Volume3,2023,100006,ISSN29497191. https://doi.org/10.1016/j.nlp.2023.100006.

[2] Akhter, Muhammad & Jiangbin, Zheng & Naqvi, Syed Irfan & Abdelmajeed, Mohammed & Zia, Tehseen. (2021). Abusive language detection from social media comments using conventional machine learning and deep learning approaches. Multimedia Systems. 28. 10.1007/s00530-021-00784-8.

[3] https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2762092.pdf.

[4] https://www.academia.edu/68659580/Abusive_language_Detection_using_Machine_Learning