Marathi Hate Speech Classification

Project Report

CS689 - Computational Linguistics for Indian Languages

Under The Guidance of Prof. Arnab Bhattacharya

Group Members

Akash Shivaji Varude (231110006)

Pankaj Siddharth Nandeshwar (231110034)

Pranjal Maroti Nandeshwar (231110035)

ACKNOWLEDGEMENT

We, as a group, would like to express our profound gratitude to Prof. Arnab Bhattacharya for his invaluable guidance and support throughout the completion of our project titled ‘Marathi Hate Speech Classification’. His mentorship has been instrumental in shaping our understanding and approach towards our research objectives.

Furthermore, we would like to acknowledge that this project was completed entirely by us and not by someone else.

We, the members of the group, extend our heartfelt thanks:

Akash Shivaji Varude
Pankaj Siddharth Nandeshwar
Pranjal Maroti Nandeshwar

ABSTRACT

Hate speech detection is a critical task in natural language processing (NLP) with significant societal impact. In this project, we focus on hate speech classification in Marathi text, aiming to develop an effective model for identifying hateful, offensive, and profane language. We leverage a custom model architecture based on XLM-RoBERTa and incorporate additional features to improve classification performance. Our approach involves tokenizing input sentences and extracting features from manually curated lists of hate, offensive, and profane words. We train the model on a balanced dataset containing examples of hate speech, offensive language, profanity, and neutral text. Through experimentation and evaluation, we demonstrate the effectiveness of our model in accurately classifying hate speech in Marathi text.

TABLE OF CONTENTS

List of Figures

1. Introduction

Hate speech, defined as speech that attacks a person or group based on attributes such as race, religion, ethnic origin, sexual orientation, disability, or gender, has become a issue in online communication. The rise of social media and online forums has made it easier for individuals to disseminate hateful and discriminatory content, leading to increased concerns about the impact of such speech on society. Detecting and mitigating hate speech online is crucial to maintaining a safe and inclusive online environment.

In this project, we address the challenge of hate speech detection in Marathi text, a language spoken predominantly in the state of Maharashtra, India. Marathi, like many other languages, faces challenges in detecting hate speech due to the nuances and complexities of language use. Existing approaches to hate speech detection often rely on manually curated lists of offensive and hateful words, which may not capture the full range of hate speech present in a given language.

To address this limitation, we propose an approach that combines a custom XLM-RoBERTa model with additional features extracted from manually curated lists of hate, offensive, and profane words in Marathi. By incorporating these features into our model, we aim to improve its ability to detect hate speech and accurately classify text into relevant categories. Through this work, we hope to contribute to the development of effective hate speech detection tools for Marathi and other languages, ultimately fostering a safer and more inclusive online environment for all users.

2. Background

Hate speech, a form of communication that aims to degrade, intimidate, or incite violence against individuals or groups based on characteristics such as race, religion, ethnicity, sexual orientation, or gender, has become a pressing societal concern. The proliferation of hate speech on digital platforms has raised alarms about its impact on social harmony, individual dignity, and democratic discourse. As awareness of the harmful effects of hate speech has grown, there has been a concerted effort to develop strategies to counter it and promote more respectful and inclusive forms of expression.

Detecting hate speech in text presents a formidable challenge due to the nuanced and context-dependent nature of language. Conventional approaches to hate speech detection often rely on simplistic keyword matching or manual annotation, which may fail to capture the subtleties of hate speech in different linguistic and cultural contexts. Consequently, there is a growing interest in developing advanced computational techniques for hate speech detection that can adapt to the complexities of diverse languages and cultural settings.

In the context of Marathi, a language spoken by millions in Maharashtra and other regions of India, the challenge of hate speech detection is particularly acute. Marathi, like many other languages, possesses its own unique linguistic features and cultural nuances that must be considered when designing hate speech detection tools. Despite the growing importance of hate speech detection, research on this topic in the Marathi language remains limited, with few studies specifically addressing this issue.

This project aims to bridge this gap by developing a hate speech classification model for Marathi text. By utilising a custom model architecture and integrating additional features extracted from lists of hate, offensive, and profane words, we aim to enhance the accuracy and effectiveness of hate speech detection in Marathi. Through this, we aspire to contribute to the broader campaign against hate speech and promote a more respectful and inclusive online discourse in Marathi and other languages.

3. Methodology

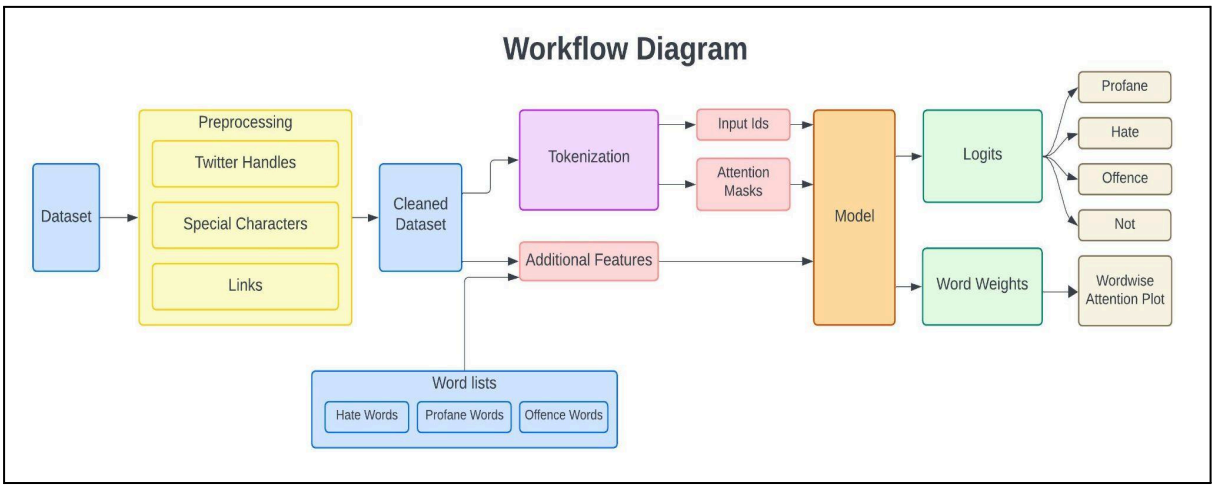

Figure 1: Workflow diagram

3.1 Dataset Collection and Preparation

We created three lists of words representing hate speech, offensive language, and profanity in Marathi. These lists were manually curated and contained words commonly used for expressing hate, offence, or profanity in the language.

The L3-cube-Mahahate dataset was used as the base dataset for this project. The text contained many trivial features like Twitter handles, special characters, urls, etc. All these along with punctuations were removed as a part of preprocessing.

3.2 Tokenization and Feature Extraction

We used the AutoTokenizer from the Transformers library to tokenize the input sentences. For each input sentence, we extracted additional features representing the presence of words from the hate, offensive, and profanity lists. These features were binary vectors indicating whether each word from the lists was present in the sentence.

3.3 Model Architecture

Our hate speech classification model is based on the XLM-RoBERTa architecture, a variant of the RoBERTa model that has been pre-trained on data. We modified the base XLM-RoBERTa model to incorporate additional features representing the presence of hate speech, offensive language, and profanity in the input sentences. Here is a detailed explanation of the model architecture:

i) Input Layer:

The input to the model consists of tokenized sentences in Marathi, generated using the AutoTokenizer from the transformers library.

Each tokenized input sentence is converted into input IDs and an attention mask, which are standard inputs for transformer-based models like XLM-RoBERTa.

ii) Additional Features:

In addition to the tokenized input, our model takes in additional features representing the presence of hate speech, offensive language, and profanity in the input sentences.

These additional features are binary vectors of the same length as the word lists for hate speech, offensive language, and profanity. Each element in the vector corresponds to whether the corresponding word from the lists is present in the input sentence.

iii) XLM-RoBERTa Base Model:

The tokenized input sentences and additional features are passed through the XLM-RoBERTa base model, which consists of multiple transformer layers.

The layers in XLM-RoBERTa encode the input sentences into contextualised representations, capturing the semantic meaning of the input text.

iv) Attention Mechanism:

We added two attention mechanisms to the model to enhance its ability to capture relevant information from the input sentences. The first attention mechanism calculates word-level attention weights, highlighting important words in the input sentences. The second attention mechanism calculates

sentence-level attention weights, aggregating the word-level representations to generate a single representation for the entire sentence.

v) Classifier Layer:

The final hidden state output from the XLM-RoBERTa base model, along with the additional features, is concatenated and passed through a classifier layer.

The classifier layer is a linear layer that maps the concatenated representation to the number of output labels (4 in our case: hate speech, offensive language, profanity, and neutral text).

The output of the classifier layer is a probability distribution over the four labels, indicating the likelihood of each label for the input sentence.

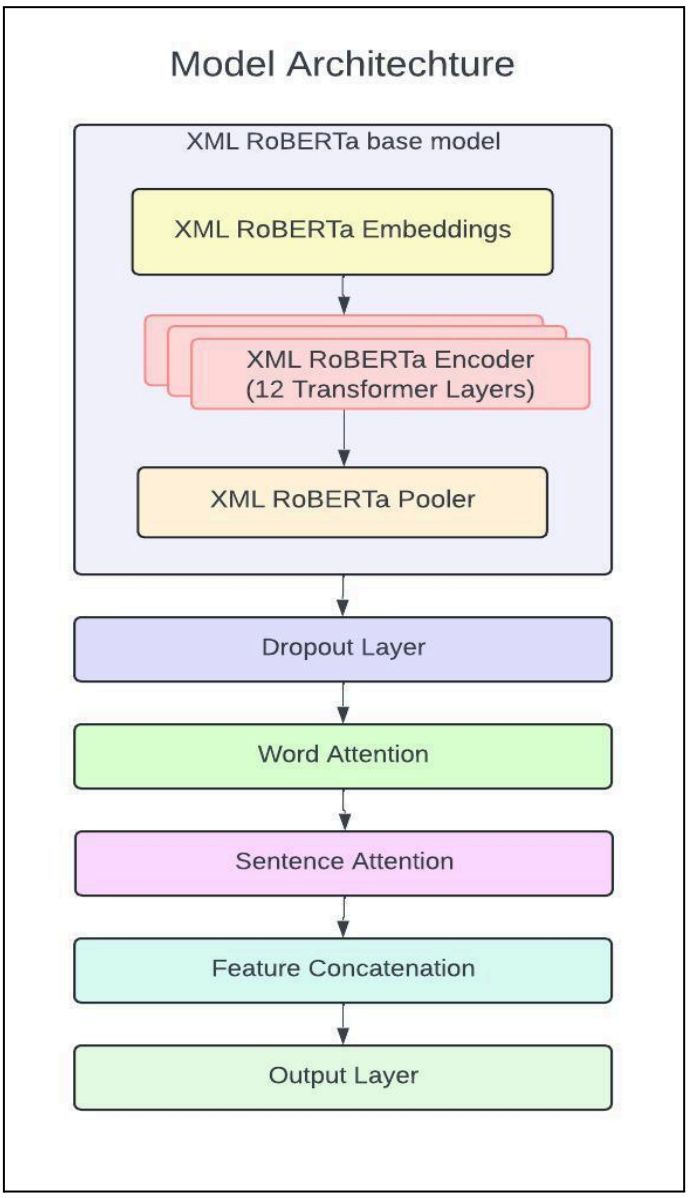

Figure 2: Model architecture

3.4 Model Training and Fine-Tuning

i) Initialization:

Our model was initialised using the pretrained XLM-RoBERTa architecture, specifically trained on multilingual data. This initialization enabled the model to benefit from the extensive knowledge learned which potentially enhanced its performance on hate speech classification in Marathi.

ii) Optimizer and Learning Rate:

For training, we utilised the AdamW optimizer, a variant of the Adam optimizer that incorporates weight decay to prevent overfitting.

The initial learning rate was set to 5e-5, a commonly used value for fine-tuning transformer-based models. This choice was based on empirical testing to optimise the model's learning rate.

ii) Batch Size:

A batch size of 8 was chosen for training. This batch size strikes a balance between computational efficiency and model performance, enabling effective training without encountering memory constraints.

iii) Loss Function:

The cross-entropy loss function was employed to compute the loss between the predicted and actual labels. This loss function is well-suited for multi-class classification tasks like hate speech detection, where the goal is to minimise the discrepancy between predicted probabilities and ground truth labels.

For a multi-class classification task with K classes, the formula is:

$$\text{Cross-Entropy Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_{i,k}\log(\hat{y}_{i,k})$$

Where,

N is the total number of examples.

K is the number of classes.

$Y_{i,k}$ is the actual label (0 or 1) for class k and example i

$\hat{Y}_{i,k}$ is the predicted probability of class k and example i

iv) Training Loop:

The model was trained for 15 epochs, with each epoch involving multiple iterations over the training dataset to learn underlying patterns. During each epoch, the model parameters were updated using backpropagation based on the computed loss over batches of data. Performance on a separate validation dataset was monitored after each epoch to prevent overfitting. Early stopping could be employed if validation loss did not improve over several epochs.

v) Evaluation:

Following training, the model's performance was evaluated on a distinct test dataset not used in training or validation. This evaluation assessed the model's ability to generalise to unseen data. Accuracy was calculated on the test dataset, representing the percentage of correctly classified examples out of the total.

vi) Model Saving:

Upon completion of training, the trained model, along with the tokenizer and other necessary configurations, was saved. This saved model could be loaded for inference on new Marathi text, enabling accurate hate speech classification.

The training and fine-tuning strategy aimed to optimise the model's performance on hate speech detection in Marathi, ensuring its robustness and effectiveness in real-world applications.

4. Results

I. After tuning the hyperparameters like learning rate using AdamW, we trained and evaluated our model.

II. The training loss we incurred was as low as 0.17 and during evaluation, we got the evaluation loss limited to 0.5. We achieved a training accuracy of 93.95% and validation accuracy of 82.8%. These were the best validation accuracy stats for the model, before the model started overfitting on the training data.

III. Test accuracy of 82.85% was achieved, marking a significant milestone in model performance and validation.

IV. Initialising the hyperparameters with the optimal values, We observed a significant enhancement in inference speed, with outputs generated swiftly, underscoring the efficiency and optimality of our model.

5. Conclusion

Our project focused on enhancing a pre-existing model for 4-class hate speech classification in Marathi text viz. Mahahate-Multi-Roberta created by L3Cube-Pune. Through the incorporation of new features to the model such as word-level attention, sentence-level attention, and a feature vector sensitive to the presence of profane, offensive, or hate-inducing Marathi words, we were able to improve upon the original model's performance. While the original authors achieved an accuracy of 78.7%, our refined model achieved an accuracy of 82.85%, which is quite an upgrade.

6. Future Scope

Moving forward, there are several avenues for further improvement. Firstly, we could explore additional linguistic features or contextual information to enhance the model's understanding of hate speech nuances in Marathi text. The list of words created by us can also be extended by including inflections of the words, and new words altogether.

Marathi, being a low-resource language, suffers from lack of datasets and thus further down the line, incorporating more diverse and expansive datasets could help generalise the model's performance across different contexts and domains. Moreover, refining the model's ability to handle nuanced linguistic constructs and subtleties specific to Marathi language could lead to even greater accuracy. Overall, our project lays a foundation for continued advancements in hate speech classification in Marathi text, with potential future enhancements to further improve performance and robustness.

7. References

1. Hrushikesh Patil, Abhishek Velankar, and Raviraj Joshi. 2022. L3Cube-MahaHate: A Tweet-based Marathi Hate Speech Detection Dataset and BERT Models. In *Proceedings of the Third Workshop on Threat, Aggression and Cyberbullying (TRAC 2022)*, pages 1–9, Gyeongju, Republic of Korea. Association for Computational Linguistics.
2. Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.
3. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 6000–6010.