MSc Project - Reflective Essay

Project Title:	Sentiment Analysis of Code-Mixed Hindi-Marathi Social Media Text in Roman Script
Student Name:	**Nachiket Kisan Shejwal**
Student Number:	**240071648**
Supervisor Name:	**Dr. Haim Dubossarsky**
Programme of Study:	Msc Big Data Science

Introduction

This project explores sentiment analysis for code-mixed Hindi–Marathi text written in Roman script, a problem with little attention in natural language processing research. It is motivated by how people increasingly communicate on digital platforms, where multilingual speakers mix Hindi and Marathi freely and prefer Roman script due to convenience, academic exposure, and professional habits. While natural for users, this blending poses challenges for computational models. The research is significant because Marathi is underrepresented in computational linguistics, and while some resources exist for Hindi, their usefulness drops when written in Roman script. Very little work addresses Romanised Hindi–Marathi mixtures, particularly in informal social media. By focusing here, the project highlights an important challenge in multilingual NLP and contributes to inclusive systems that reflect real-world linguistic behavior.

A dataset was created using comments from social media, YouTube, X posts, and regional newspapers. These sources reflect how users naturally switch between Hindi and Marathi while writing in Roman script. The dataset captured switching patterns within a sentence, across sentences, or through short tags and expressions. Such variation ensured it represented the richness of real communication rather than simplified examples. Preparing the dataset required preprocessing. Roman script text was cleaned to remove symbols, numbers, and links. It was then transliterated into Devanagari, a shared script for both Hindi and Marathi, giving a consistent representation for models. This introduced semantic ambiguity, since some words share spelling but differ in meaning. For instance, “kaal” means “yesterday” in Marathi but “bad time” in Hindi, while “sahi” means “signature” in Marathi but “correct” in Hindi. Handling such cases was crucial to avoid misinterpreting sentiment.

The development pipeline tested how well sentiment classification methods adapt to this code-mixed, script-shifted setting. Several experiments were conducted with different approaches, parameters, and dataset versions. The outcomes gave insights into challenges of low-resource, code-mixed text and showed how well methods generalize across scripts. Beyond technical contributions, the project reflects broader cultural and social dimensions. By focusing on Hindi–Marathi, it represents millions of speakers who fluidly navigate multilingualism and adopt Roman script for convenience and accessibility. Designing systems that interpret such communication is essential for NLP to stay relevant in multilingual, digital contexts.

This project is both a technical exploration and a step toward linguistic inclusivity. It demonstrates dataset creation in challenging settings, preprocessing to address ambiguity and script mismatches, and applying computational methods to real user communication instead of idealized monolingual text. In doing so, it underlines the need to extend research beyond resource-rich languages and scripts, ensuring advances in technology support communities that use language in diverse and complex ways.

Strengths and Weaknesses

One of the main strengths of this project lies in the way the dataset was created. Instead of relying on pre-existing corpora, which are often unavailable for low-resource languages like Marathi, the dataset was constructed by drawing material from different real-world sources. Social media comments, YouTube comments, and online newspapers all formed part of the collection. This variety made the dataset more representative of actual communication patterns, especially since speakers frequently move between formal and informal registers depending on context. A further strength was the deliberate inclusion of intra-sentential, inter-sentential, and tag-switching examples. By covering these three dimensions, the dataset reflected the true complexity of code-mixed conversations. This choice made the models not only more robust but also better at generalising, since the training data captured most of the ways in which speakers combine Hindi and Marathi in practice.

Another strength was the implementation of a transliteration pipeline that converted Roman script into Devanagari. This step made it possible to process the data in a way that could work across both languages without needing to explicitly separate Hindi words from Marathi ones. It reduced the burden of manual classification while still allowing deeper processing. The project also benefited from a broad evaluation framework. Four different transformer-based models were implemented and tested, which provided a comparative understanding of their strengths and weaknesses. This multi-model implementation ensured that the analysis did not rely too heavily on a single architecture, but instead reflected a range of possibilities within transformer-based approaches. The use of K-Fold cross-validation added further strength, as it allowed the findings to be more reliable by testing the models across several partitions rather than relying on a single train-test split. Another notable strength was the inclusion of cross-script testing. By training on Roman script and testing on Devanagari, and vice versa, the project addressed a practical reality of how users communicate online. Many users freely switch between scripts depending on the platform, and this evaluation showed how resilient or fragile the models were to such variation. A further positive element was the decision to test models separately on each type of sentence (intra-sentential, inter-sentential, and tag-switching) after creating the dataset. This helped in observing whether any model behaved differently depending on the specific structure of the input. For example, some models found it easier to handle intra-sentential code-mixing while struggling more with tag-switching. Identifying these differences gave a deeper insight into the strengths and limitations of transformer architecture when applied to mixed-lingual texts. Finally, personal strength contributed to the overall project. Having strong command over both Marathi and Hindi not only made it easier to design and annotate the dataset but also helped in identifying subtle linguistic ambiguities that an outsider might have overlooked.

Despite these strengths, there were also limitations that shaped the outcomes. The most important weakness was the restricted size of the dataset. Although multiple sources were used, the amount of data collected was still modest compared to the vast corpora available in high-resource languages. A small dataset inevitably limits how well models can generalise, and this was evident in cases where performance fluctuated depending on the nature of the input. Another challenge came from transliteration. While the pipeline allowed Roman text to be mapped to Devanagari, it was not always perfect. Ambiguities in meaning made it hard to capture nuance. For example, the word "*kaal*" can mean "yesterday" in Marathi but "bad time" in Hindi, while "*sahi*" means "signature" in Marathi and "correct" in Hindi. Handling such cases required extra care and sometimes manual intervention, which slowed down the process. There were also resource-related weaknesses. The project relied on limited computational infrastructure, which meant that experimenting with very large models or running extensive hyperparameter searches was not always feasible. Fine-tuning deep transformer models demands significant GPU power, and time constraints often made it difficult to run multiple experiments in parallel.

Similarly, hyperparameter tuning, while attempted, could not be explored as systematically as planned because of these resource limitations. Debugging multiple models also required a lot of time and patience, as errors often arose due to differences in tokenization, preprocessing, or training pipelines. This slowed down progress and sometimes forced compromises in how thoroughly each model could be tested.

Possibilities for Further Work

Although this project successfully implemented a complete pipeline for sentiment analysis of Hindi–Marathi code-mixed text, there are still several areas that can be explored further to strengthen its outcomes and extend its impact. These possibilities emerge both from the limitations encountered during the research and from the opportunities highlighted in the evaluation stage. One of the key areas of future work involves building a larger and more diverse annotated dataset. While the current dataset is carefully curated from social media comments, WhatsApp chats, and newspapers, its size remains limited compared to high-resource language datasets. Expanding it through systematic collection or even crowdsourcing could provide better coverage of dialects, styles, and real-world variations in code-mixed communication. This would help in reducing model bias and making the system more representative of everyday online discourse.

Another important extension would be to refine the handling of transliteration ambiguity. During the project, challenges arose with words like *“kaal”* and *“sahi”*, which carry different meanings in Hindi and Marathi depending on the context. While preprocessing and transliteration helped reduce noise, future work should focus on addressing this challenge at the core level by explicitly identifying ambiguous words and then predicting whether their meaning belongs to Hindi or Marathi. At present, context-awareness provides some support, but more systematic investigation is needed to address these issues comprehensively. Future work will require deeper research on ambiguity-level separation at the linguistic level. From a modelling perspective, further work could involve experimenting with advanced training strategies. This includes more systematic hyperparameter tuning using techniques such as Optuna or Grid Search, which would provide deeper insights into optimal configurations for each transformer model. Similarly, additional K-Fold cross-validation experiments could be carried out with more folds or stratified sampling to strengthen the reliability of evaluation. Another possibility is to test ensemble strategies, combining the strengths of multiple models to achieve greater accuracy and robustness.

Future work can also extend to real-world applications. One direction would be developing a real-time sentiment analysis tool capable of handling live social media data streams. Such a system could provide immediate insights into public opinion for businesses, policymakers, or media organizations. In addition, testing robustness against noisy social media text, such as spelling mistakes, slang, and emojis—would make the system more practical for deployment in dynamic, user-generated environments. This research could also benefit from making the model’s decisions easier to understand. If we can see which words the model pays attention to when predicting sentiment, it will give us more clarity about how the results are produced. This would not only help improve the model but also make people trust its predictions more.

Work Possible with More Time

If I had more time to work on this project, I would have first addressed an important issue I observed during transliteration from Roman script to Devanagari script. Occasionally, diacritic errors appeared when words with similar pronunciation were converted incorrectly, making them grammatically inaccurate in the target script. While the overall transliteration pipeline worked well, these errors reduced the precision of the text quality and could affect downstream sentiment prediction. With additional time, I would have explored ways to minimize such errors, ensuring more accurate script conversion and thereby improving the reliability of the whole pipeline. I would also have dedicated more effort towards strengthening the accuracy of the results. Although the models achieved good performance, achieving higher precision and consistency would have been possible with extended experimentation. Extra time would have allowed me to conduct more rigorous hyperparameter tuning, refine preprocessing methods, and possibly integrate additional linguistic features, all of which could have pushed the models to stronger outcomes.

Another direction I would have liked to take is practical implementation. My vision was to create a real-time application, perhaps a simple website using Flask or a similar framework, where users could input their messages and instantly receive sentiment predictions. Such a tool would not only demonstrate the applicability of the research but also provide an accessible way for others to interact with and benefit from the system. Finally, I believe that expanding beyond binary sentiment categories would have made the project more powerful. At present, the focus has been on positive and negative classification, but real human expressions often carry subtler tones. Including categories like *strong positive, positive, neutral, negative,* and *strong negative* would add greater depth. For instance, detecting strongly negative sentiment in real-time chats could help in identifying potentially harmful situations. Such granularity would better reflect the complexity of real-world communication and expand the usefulness of this research in applied contexts.

Analysis of the relationship between theory and the practical work produce

From a theoretical perspective, transliteration was expected to act as a straightforward bridge between Roman script and Devanagari, allowing the models to process text without ambiguity. In practice, this pipeline was indeed successful since both Hindi and Marathi share the same script, and the implemented transliteration step converted not only Hindi–Marathi tokens but also English words into Devanagari script. This ensured uniformity in the dataset and avoided the inconsistency that might have arisen from leaving English tokens untouched. The result was closer to theoretical expectations, although it also showed that transliteration itself required fine-tuned preprocessing to handle real-world inputs.

Another theoretical assumption was that larger multilingual models such as *XLM-R* and **mBERT**, trained on 100+ languages, would have a natural advantage because of their higher range of tokenized corpus. This suggested that they would carry broader vocabulary coverage and stronger generalisation capacity across languages. However, in practice, the *MuRIL* model consistently outperformed others. Its focus on Indian languages and its training on transliterated Indic corpora made it more suitable for this project. The superiority of *MuRIL* was not limited to the general results but was observed across all types of data created in the project, including intra-sentential, inter-sentential, and tag-switch code-mixing. Even in advanced evaluation, where models were tested separately on each sentence type, *MuRIL* showed a robustness that theory did not entirely predict.

The relationship between theory and practice was also highlighted in model evaluation. Theoretical guidance often treats accuracy as the primary measure of success, but in this project, it became clear that accuracy alone could not reflect the models' behaviour. Using Stratified K-Fold cross-validation and classification reports (precision, recall, and F1-score), it was found that performance varied not only across models but also across sentence types. For instance, *XLM-R* and *IndicBERT* occasionally performed well in certain mixing scenarios but dropped in others, while *MuRIL* remained more consistent. This showed that theoretical expectations of uniform transformer performance did not fully hold in practice; the behaviour was more nuanced and depended heavily on the type of input data.

Awareness of Legal, Social, Ethical, and Sustainability issues

This project brings together language, technology, and everyday human communication, and in doing so, it naturally raises important legal, social, ethical, and sustainability questions alongside the technical work.

Legal

The data was gathered from openly accessible sources (e.g., public social media comments and online news) and processed into a research dataset. No personal identifiers (such as names, phone numbers, email addresses, profile links) were intentionally retained in the working data, and the pipeline focuses on text only. When using material from newspapers, the content was treated for research/analysis rather than redistribution; full-text reproduction was avoided, and sources are cited in the report. For any public-message style samples (e.g., X-post), the principle followed avoid hate rate content and bias towards any particular region, religion or political party: use only text that you have a right to use, store it without identifiers, and keep it for the minimum time needed for analysis. All pre-trained models and libraries are used under their published licenses and are credited in the report. If the dataset or trained models are shared in future, they should be released with a clear licence, a short usage notice ("for research only; do not use to profile individuals"), and a brief data statement describing provenance, preprocessing, and known limitations.

Social.

The project aims to support Hindi–Marathi speakers whose real online writing is code-mixed and often typed in Roman script. That choice promotes inclusivity for a community under-served by standard, monolingual NLP resources. At the same time, representativeness matters: code-mix style varies by region, age, and platform. To reduce social bias, the dataset was built from multiple sources and the evaluation explicitly tested three common code-mix patterns (intra-sentential, inter-sentential, tag-switch). Reporting results by type makes it easier to see if any group or style is being disadvantaged. If deployed, the system should avoid being framed as a tool to "correct" language; instead, it should recognise mixed Hindi–Marathi as a valid way people communicate online.

Ethical

Two main risks are misclassification and misuse. Short or slang-heavy posts can be wrongly classified, but this project reduced that risk by using K-Fold cross-validation, classification reports, per-type testing, and cross-script checks, which made errors easier to see. Misuse is another concern, as sentiment models could be used to profile or suppress speech. To address this, the project only used data for research and aggregate analysis, avoided collecting personal details, and highlighted the system's limits. If a future demo is built, it should have a clear privacy notice, avoid logging user inputs by default, and give users an opt-out. Any "strong negative" flags should be used as supportive signals with human oversight, not as final judgements.

Sustainability

Instead of training large models from scratch, this project fine-tunes existing ones, reuses artefacts like tokenizers and checkpoints, and keeps hyperparameter searches limited. This makes the work easier to extend in future research and helps in building more sustainable solutions. If deployed, smaller or optimised versions of the models could make the system more efficient and practical, supporting long-term applications for human welfare.