Innovations in Code-Mixed Hate Speech Detection: The LLM Perspective

Sargun Nagpal
Center for Data Science
New York University
sn3250@nyu.edu

Sharad Dargan
Center for Data Science
New York University
sd5251@nyu.edu

Harsha Koneru
Center for Data Science
New York University
hk3820@nyu.edu

Shikhar Rastogi
Center for Data Science
New York University
sr6644@nyu.edu

Abstract

Detecting and flagging hate speech and abuse on social media platforms is an important and time-sensitive task. While supervised learning approaches have been successful in identifying hate speech in English and some other high-resource languages, this is not the case for code-mixed text, which is a common way of communication for many bilingual people. In this project, we evaluate the effectiveness of Large Language Models for the task of Hindi-English code mixed hate speech detection, and compare this to existing BERT-based models on an existing "Hinglish" Indian Politics hate speech dataset. Additionally, we evaluate the generalization capabilities of these models on a custom Hindi-English code-mixed hate speech dataset. We find that smaller specialized finetuned models such as Hing-RoBERTa outperform both prompted and finetuned LLaMa-2 on the existing Hinglish Indian Politics dataset, and also generalize better to our newly collected dataset.

1 Introduction

Code-mixing, a phenomenon where people in mix multiple languages interchangeably during conversation, is frequently observed in many multilingual communities. Unlike pure monolingual and multilingual tasks where the performance of Large Language Models (LLMs) has been extensively evaluated [1][2], there is a dearth of rigorous studies on the performance of LLMs in code-mixed language understanding. This is primarily due to data scarcity and the high expenses and time potentially incurred in collecting and annotating code-mixed data.

The project aims to study the effectiveness of LLMs for the task of Hindi-English code mixed hate detection and compare this to existing BERT-based models. For this task, we use an existing hate speech dataset comprising of 8500 "Hinglish" tweets on Indian Politics[3]. We additionally collect Hindi-English code-mixed hate speech data on domains such as gender, religion and sexual orientation to test the generalization capabilities of the above models.

Our results indicate that despite multilingual LLMs exhibiting promising outcomes in certain tasks using zero or few-shot prompting, they still underperform in comparison to fine-tuned models of much smaller scales. We find that current Current large language models (LLMs) exhibit lower proficiency in handling code-mixed text compared to their performance in processing English and multilingual text, and there is a need for further research to change this.

Code: https://github.com/shikharras/cm-hate-speech-detection

2 Related Work

There has been considerable interest in detecting hate speech and toxic language using NLP. Arco et al. (2023)[4] conducted a comprehensive evaluation of hate speech detection approaches, comparing fine-tuned models, Zero Shot Learning with language models, and commercial API solutions. Their findings suggest that instruction fine-tuned models with prompting techniques perform on par with or even outperform fine-tuned models. However, they did not explore code-mixed datasets, leaving a gap in understanding potential challenges in hate speech detection in code-mixed contexts.

On the other hand, Zhang et al. (2023)[5] explored the capabilities of multilingual LLMs for code-switching tasks, noting that existing models under-perform compared to smaller fine-tuned models due to the lack of explicit pretraining objectives for code-mixed data. However, their evaluation did not extend to hate speech detection, and focused on tasks like sentiment analysis, and summarization. Our study is an amalgam of the two studies and aims to bridge this gap by exploring the capabilities of LLMs for code-mixed hate speech detection.

Das et al. (2023)[6] highlight recent progress in enhancing pretraining for code-mixed models. This includes the development of novel pretraining objectives, such as SwitchMLM, which involves masking tokens at language boundaries. Architectural changes, like adding residual connections to propagate language-level information, have also been explored. These advancements in code-mixed NLP models show considerable potential.

3 Approach

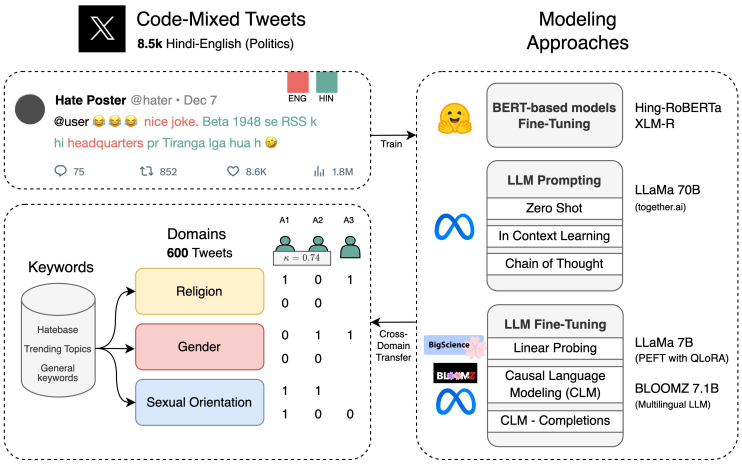

Figure 1: Overview of our approach

Data Curation: We describe the data curation steps in Section 4.1.
Modeling: We trained our models on the politics domain dataset and evaluated their cross-domain performance on our custom dataset. Our approaches include fine-tuning BERT-based models like HingRoBERTa[7] and XLM-RoBERTa, LLM Prompting in zero-shot (ZSL), in-context (ICL) and chain of thought (CoT) settings, and LLM Fine-tuning of LLaMa2 7B[8] and BLOOMZ models in classification and causal language modeling scenarios. This involved generating the entire sequence and generating only the output label (completion-only).

4 Experiments

4.1 Data

1. **Primary Dataset (Hinglish 8.5k):** We used the offensive tweets dataset [3] containing 4,612 English monolingual and 3,900 Hindi-English Indian politics themed code-mixed tweets. Each entry in this dataset consists of a tweet ID paired with its respective label - offensive (OFF) or not offensive (NOT).

2. **Supplementary Dataset (OOD):** To facilitate cross-domain evaluation and analysis, we curated a smaller dataset comprising approximately 600 Hindi-English code-mixed tweets using keywords, hate-terms from Hatebase[9] and Twitter's trending topics[10]. Our goal with this dataset is to represent a broader spectrum of contexts including religion, gender and sexual orientation. We refined the dataset by filtering out tweets in languages other than English and English-Hindi codemixed, eliminating duplicate tweets, and ensuring the removal of any personally identifiable information such as usernames and pictures. We label these tweets manually with the methodology aligned with the protocol established by Nafis et al. 2023 [11]. The average Inter-Annotator Agreement Score (Cohen's Kappa) is 0.74 across various domains. Given the subjective nature of hate speech, we have openly released all annotations for potential utilization in future research.

4.2 Evaluation method

Quantitative Evaluation: We assessed the performance of our models on the binary classification task by computing the macro F1 scores to account for both the precision and recall of each class.

Qualitative Evaluation: To gain a deeper understanding of our models' performance, we conducted a qualitative error analysis by examining misclassified examples and identified common error patterns for each model. We also identified the failure modes associated with LLM prompting techniques.

4.3 Experimental details

Fine-tuning BERT-Based Models: Utilizing HuggingFace, we fine-tuned BERT-Based models for 3 epochs, employing a learning rate of 2e-5 and a batch size of 32. Our fine-tuning process involved two variations of the HingRoBERTa model: HingRoBERTa trained solely on the Roman script and HingRoBERTa-mixed trained on both Roman and Devnagari scripts.

LLM Prompting - Chain of Thought (CoT): Within the LLM prompting methodology, particularly the CoT approach [12], our prompts included a definition of offensive speech alongside approximately ten examples. Each example comprised an Input tweet, a justification for the classification, and a correct label. This technique aimed to encourage the model to generate tokens to 'reason' about the problem, providing context and supporting information for hate speech classification.

LLM Fine-tuning: Classification with Linear Probing was conducted by freezing all model parameters except the last linear layer. This model was trained with a learning rate of 5e-4 for 10 epochs. The Causal Language Modeling LLaMa model was trained by employing Parameter Efficient Fine-Tuning (PEFT) utilizing NF4 Quantization and Low-Rank Adaptation (LoRA) [13] with rank 64 matrices. The trl library was used to create a response template to mask the prompt tokens for completion-only language modeling.

4.4 Results

The results of our study comparing LLMs and fine-tuned models for detecting code-mixed hate speech indicate comparable performance between LLMs and significantly smaller fine-tuned models. Within LLMs, the investigation into different prompting methods revealed noteworthy observations: Chain of Thought (CoT) outperformed Zero-Shot and In-context Learning (ICL), allowing LLMs to generate tokens to deliberate on the problem. ICL, however, demonstrated sensitivity to example choice and manifested various biases. When framing hate speech detection as a binary label prediction task, approaching it as a generative problem (CLM), yielded superior results compared to classification. Additionally, our study found that prompting with the 70B LLaMa model performed similarly or worse than fine-tuning the smaller 7B model. With CoT prompting, we gain insight into the reasoning behind the model's predictions but do not get a probability distribution over the class labels. Regarding model generalization, the HingRoBERTa model exhibited the best generalizability, closely followed by the LLaMa 7B CLM model. A scatter plot showing the generalization capacity of the model finetuned on the Hinglish dataset and evaluated on the OOD Dataset is shown in Figure 2

Table 1: Results

	BERT Based Models Finetuning			LLM Prompting			LLM Finetuning			
Metric (Macro-F1)	Hing RoBERTa	Hing RoBERTa Mixed	XLM-R	LLaMa 70B ZSL	LLaMa 70B ICL	LLaMa 70B CoT	LLaMa 7B Linear Probing	LLaMa 7B CLM	LLaMa 7B CLM - Completion Only	BLOOMZ 7.1B CLM - Completion Only
Hinglish 8.5K Dataset										
Monolingual	76.74	**79.92**	19.88	53.47	43.24	69.38	72.44	67.47	75.26	75.01
Code-Mixed	**81.97**	79.64	26.13	52.44	42.33	70.42	72.31	58.41	78.87	75.13
Overall	80.12	**80.15**	23.12	53.36	43.4	70.34	72.62	62.71	77.53	75.37
Out of Domain (OOD) Dataset (Inference Only)										
Gender	78.05	**78.77**	31.35	48.27	36.61	54.87	50.68	64.06	74.45	58.57
Religion	73.82	71.58	28.21	54.9	36.33	63.02	61.55	49.65	**77.08**	67.56
Sexual Orientation	67.59	**73.48**	32.48	42.5	32.48	55.83	52.21	60.65	64.38	62.26
Overall	74.88	**75.15**	30.32	49.93	36.05	58.9	55.53	59.11	74.23	62.85

5 Analysis

We see in Table 2, that there are some common themes where the LLM-based approaches go wrong. While the chat-based models are overly cautious, the LLM finetuning-based approaches fail due to an inadequate understanding of the context and grammar.

Table 2: Tweet Evaluation Results

Dataset	Tweet Text	Model	Error	Reason
OOD	Matlab sirf ladki ke character baat ithae tab bologe 0ar ladke ke upar wo bhi khud ke fd se karoge to chup rahoge	LLaMa-70b-Chat CoT	False Positive	Overly cautious due to alignment as a chat model
Hinglish 8.5K	Ab kaun mcd Hindu-Muslim kar rha	LLaMa-7b CLM CO	False Negative	Different ways of spelling the same thing
Hinglish 8.5K	Marathi nhi Hindu... Marathi ke wajah se nhi Hindu hone ke wajah se Target kiya... musalman hota to nhi hota	LLaMa-7b CLM CO	False Negative	Lack of nuanced understanding (No profane words)
Hinglish 8.5K	Jaisa tumko Muslim atanki nhi pasand... Waise humko Hindu atanki nhi pasand... Hum bhi bhot se Hinduo ki izzat kartey hai, Mager jahilo ki nhi...	LLaMa-7b CLM CO	False Positive	Use of negative connotation words but not offensive

6 Conclusion

We find that smaller fine-tuned models pretrained on more relevant data trump general LLMs on this niche task of Hindi-English hate speech detection. Also, we see that prompting LLMs with CoT biases the models heavily toward the examples and justifications in the prompt - and this is not suitable for highly subjective tasks. Smaller BERT-based models generalize better than both prompted and fine-tuned LLMs. In the future, this work can be extended to evaluate LLMs capable of handling Indian languages and cultural contexts such as the newly released OpenHathi-7B.

References

[1] Haoyang Huang, Tianyi Tang, Dongdong Zhang, Wayne Xin Zhao, Ting Song, Yan Xia, and Furu Wei. Not all languages are created equal in llms: Improving multilingual capability by cross-lingual-thought prompting. *arXiv preprint arXiv:2305.07004*, 2023.

[2] Kabir Ahuja, Rishav Hada, Millicent Ochieng, Prachi Jain, Harshita Diddee, Samuel Maina, Tanuja Ganu, Sameer Segal, Maxamed Axmed, Kalika Bali, et al. Mega: Multilingual evaluation of generative ai. *arXiv preprint arXiv:2303.12528*, 2023.

[3] Nazia Nafis. Towards safer communities: Detecting aggression and offensive language in code-mixed tweets to combat cyberbullying. `https://github.com/surrey-nlp/woah-aggression-detection/blob/main/data/New10kData/cyberbullying_10k.csv`, 2023.

[4] Flor Miriam Plaza-del arco, Debora Nozza, and Dirk Hovy. Respectful or toxic? using zero-shot learning with language models to detect hate speech. In *The 7th Workshop on Online Abuse and Harms (WOAH)*, pages 60–68, Toronto, Canada, July 2023. Association for Computational Linguistics.

[5] Ruochen Zhang, Samuel Cahyawijaya, Jan Christian Blaise Cruz, Genta Indra Winata, and Alham Fikri Aji. Multilingual large language models are not (yet) code-switchers, 2023.

[6] Richeek Das, Sahasra Ranjan, Shreya Pathak, and Preethi Jyothi. Improving pretraining techniques for code-switched NLP. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1176–1191, Toronto, Canada, July 2023. Association for Computational Linguistics.

[7] Ravindra Nayak and Raviraj Joshi. L3Cube-HingCorpus and HingBERT: A code mixed Hindi-English dataset and BERT language models. In Girish Nath Jha, Sobha L., Kalika Bali, and Atul Kr. Ojha, editors, *Proceedings of the WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference*, pages 7–12, Marseille, France, June 2022. European Language Resources Association.

[8] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[9] Hatebase. `https://hatebase.org/`. Accessed: 2023-12-15.

[10] Twitter trending topics archive. `https://archive.twitter-trending.com/`. Accessed: 2023-12-15.

[11] Nazia Nafis, Diptesh Kanojia, Naveen Saini, and Rudra Murthy. Towards safer communities: Detecting aggression and offensive language in code-mixed tweets to combat cyberbullying. In *The 7th Workshop on Online Abuse and Harms (WOAH)*, pages 29–41, Toronto, Canada, July 2023. Association for Computational Linguistics.

[12] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

[13] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

7 Appendix

7.1 Generalization Capabilities

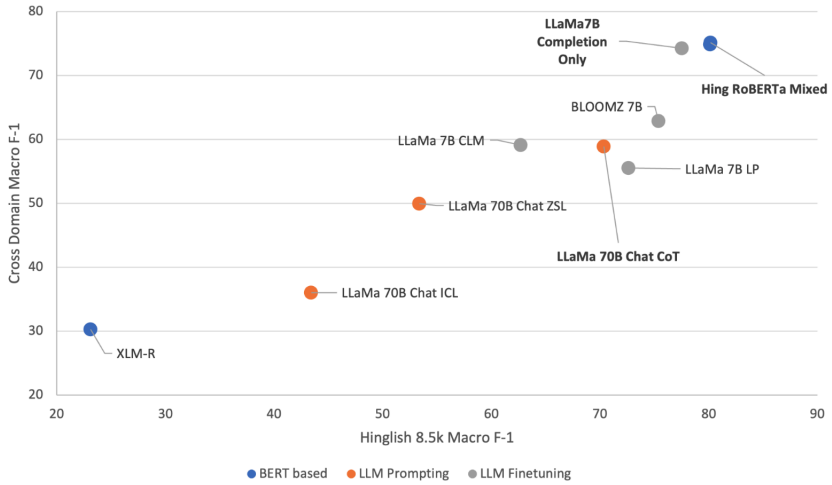

Figure 2: Generalization Capabilities

7.2 Code

The code for the project is available on `https://github.com/shikharras/cm-hate-speech-detection`