DweshVaani: An LLM for Detecting Religious Hate Speech in Code-Mixed Hindi-English

Varad Srivastava
Barclays
varadsrivastava.iitdelhi@gmail.com

Abstract

Traditional language models in NLP have been extensively made use of, in hateful speech detection problems. With the growth of social media, content in regional languages has grown exponentially. However, use of language models as well as LLMs on code-mixed Hindi-English hateful speech detection is under-explored. Our work addresses this gap by investigating both cutting-edge LLMs by Meta, Google, OpenAI, Nvidia as well as Indic-LLMs like Sarvam, Indic-Gemma, and Airavata on hateful speech detection in code-mixed Hindi-English languages in a comprehensive set of few-shot scenarios which include examples selected randomly, as well as with retrieval-augmented generation (RAG) based on MuRIL language model. We observed that Indic-LLMs which are instruction tuned on Indian content fall behind on the task. We also experimented with fine-tuning approaches, where we use knowledge-distillation based-finetuning by using rationale behind hate speech. Finally, we also propose Dwesh-Vaani, an LLM based on fine-tuned Gemma-2, that out-performs all other approaches at the task of religious hateful speech detection as well as targeted religion identification in code-mixed Hindi-English languages. Black-box functionality testing is used to establish its robustness and stability.

1 Introduction

Due to the exponential growth of internet, social media has become a preferred medium for individuals to share their views and thoughts. According to some estimates, more than half of the world now actively use social media, with this number growing by around 260 million, just over the last year. While social media has undoubtedly brought people closer and made access to information easy, it has also been used over the years to spread hate and misinformation. Social media is used by individuals with malicious intents to spread hate by amplifying their stereotypical and derogatory views expressing hate or encouraging violence at an individual or community (based on their race, religion, sex etc.), which can reach audience in all corners of the world in a matter of seconds. The presence and generation of such negative contents can create divide within individuals and communities within society and lead to conflicts and hostility. Previous studies have also pointed out the urgency to address this issue, in order to maintain a peaceful and inclusive society.

Over the past few years, numerous research studies have examined different aspects of hate speech and offensive language detection, including related topics such as identifying abusive content (Nobata et al., 2016; Sazzed, 2021), cyberbullying detection (Paul et al., 2022; Iwendi et al., 2020) and detecting hostility (Bagora et al., 2022; Kamal et al., 2021; Raha et al., 2021). Some of these studies have specifically concentrated on targeted hate speech (Chiril et al., 2021; ElSherief et al., 2018; Sharma et al., 2023), while others have explored issues related to vulnerable communities, such as detecting homophobia and transphobia (Sharma et al., 2022; Chakravarthi et al., 2022)

In spite of recent research, the incidences of religious hate speech are continuously rising every day. Additionally, we have started to see drastic real-world consequences of hate speech targeting religious groups, one prominent example being the Rohingya crisis in Myanmar, where social media was used extensively to spread misinformation and incite violence against the minority Rohingya Muslim community. Similarly, anti-Semitic hate speech surged across the social media during

Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025), pages 46–60
January 19, 2025.

the conflict between Hamas and Israel, inciting violence. Given the huge volumes of hate speech content that is generated on social media, language models have been used extensively in previous studies for automatic detection of harmful content.

Most of the existing research works focus on content in English language, and there is relatively less study on hateful content on social media in low-resource languages like Hindi. Hindi is spoken across 20 countries with 577 million native speakers and now has a significant amount of content available online, as people are increasingly expressing themselves on social media in regional or code-mixed languages. However, lack of suitable datasets for targeted hate speech against religion in Hindi or Hindi-English code-mixed languages, had rendered the progress slow. Recently, Targeted Hate Speech Against Religion (THAR) dataset (Sharma et al., 2024) was curated, in a step towards building language models to bridge this gap. Additionally, given the language understanding prowess of LLMs, they have been under-explored on tasks like hate speech detection, with a scarcity of research in this area.

In this paper, we bridge the aforementioned gap by evaluating both Multi-lingual Masked Language Model (like MuRIL) and few-shot learning techniques with cutting-edge LLMs like Llama-3.1, Gemma-2, and GPT-4o Mini as well as Indic LLMs like Sarvam, Nemotron, Airavata, and IndicGemma.

Therefore, we perform analyses and evaluations in following four different approaches. Firstly, we evaluate a multilingual masked language model on the task of religious hate speech detection in YouTube comments and identification of religion that the comment hatefully targets.

Secondly, we make use of in-context learning using state-of-the-art open-source Instruction-tuned LLMs, to evaluate their performance on zero-shot and few-shot (1 and 2 examples per class) learning and Retrieval-Augmented Generation (RAG). Thirdly, we experiment with fine-tuning and knowledge-distillation approaches using the LLMs.

Finally, we propose DweshVaani, an LLM based on Gemma-2, that can be used as state-of-the-art model for religious hate speech detection in Hindi and Hindi-English code-mixed languages. We show that this model is able to outperform all other approaches, as well black-box testing shows it to be more robust. Additionally, we also perform extensive error analysis of the best performing model.

2 Related Work

2.1 Hate-Speech Detection

Previous studies have focused on different facets of hate speech detection, like abusive comments (Nobata et al., 2016; Sazzed, 2021; Zia Ur Rehman et al., 2023), offensive content (Salaam et al., 2022; Saumya et al., 2021; Chen et al., 2012), toxic speech (D'Sa et al., 2020; Nguyen et al., 2021), including transphobia (Chakravarthi et al., 2022; Sharma et al., 2022) and misogyny detection (Nozza et al., 2019; Pamungkas et al., 2020). Hate speech concerning religion is a sensitive topic as it has the power to create divide in societies, as recently observed in the Delhi riots. There do exist studies on religious hate speech detection but have often not been comprehensive in covering major religions, for example, some cover only Islamophobia detection (Mehmood et al., 2021). Due to lack of suitable datasets and generalized models, religious hate speech detection in low-resource languages remains under-explored.

Despite a significant amount of population speaking Hindi, datasets for religious hate-speech detection in Hindi or code-mixed Hindi-English were largely absent. Targeted Hate Speech Against Religion (THAR) dataset (Sharma et al., 2024) was recently released which aimed to address this gap. It not only has information about whether a comment was hatefully targeting a religion, but also encompasses information about which religion it was targeting. We make use of this dataset in our study.

2.2 Using LLMs for Hate Speech Detection

There has been some focus on using LLMs to generate datasets on hateful speech. Hartvigsen et al. (2022) used demonstration-based prompting for LLMs, to encourage it to generate both toxic and benign sentences that talk about minority groups without any sort of explicit words or language. In this way they

created the TOXIGEN dataset which encompasses implicit toxicity which can be used to train language models to detect subtle toxicity rather than getting confised with any mentions of minorities. Some works like that by Das et al. (2022a) have even constructed evaluation sets (HateCheckHIn) in Hindi for testing multilingual functionalities of hate speech detection models. In a further work, Das et al. (2024) evaluated ChatGPT 3.5 on these functionality tests and observed that it performs inferior on Hindi as compared to other languages.

Guo et al. (2023) assessed the ability of LLMs in hate speech detection by employing them on five datasets, namely - HateXplain, COVID-HATE, CallMeSexist, USElectionHate and SWSR. They used few-shot learning and chain-of-thought prompting techniques with GPT3.5-Turbo along with fine-tuned BERT, and RoBERTa models, and observed that ChatGPT consistently out-performs both of them. Additionally, when they tested ChatGPT on multilingual hate speech detection in Chinese language, it was observed that ChatGPT performs significantly poor than expected. Roy et al. (2023) evaluated GPT3.5, flan-T5-large and text-davinci, across three datasets - HateXplain, implicit hate, and Toxic-Spans, which contain the ground truth explanations as well. Using vanilla prompts, flan-T5-large came out to be the best performing model among the three. Additionally, when the prompt included information about the target community, performance gains of upto 30% are obtained. Sen et al. (2024) explored use of Tiny LLMs like TinyLlama, phi-2 and opt-1.3B on two hatespeech datasets, namely - DynaHate and hateeval. They observed significant gains in performance across all models for both datasets, when they fine-tuned the models using LoRa, with opt-1.3b model coming out as the best performing model.

These works have multiple gaps - they do not either assess LLMs on religious hateful speech in code-mixed Hindi-English languages, or do not present the targeted religion, which is a more challenging task than hate identification.

3 Task and Dataset

Hateful speech detection in Code-mixed Hindi-English languages is a challenging problem because of scarcity of appropriate datasets. For this work, we used the recently released Targeted Hate Speech Against Religion (THAR) dataset (Sharma et al., 2024). It consists of comments from YouTube videos scraped from videos discussing controversial topics in religious contexts, including political discussions. Two sub-tasks are proposed by the creators for this dataset:

Subtask 1 (Binary Classification): In this, the comments are classified into two categories - Anti-religion and Non-Anti-religion.

1. *Anti-religion*: A comment falls under the anti-religion class if it meets one or more of the following criteria:
    - Show hostility towards religious beliefs and their sacred elements.
    - Attack or belittle any religious faith.
    - Critique the practices and rituals associated with a specific religion.
    - Exhibit hate towards spiritual leaders or celebrities who promote a particular religion and its cultural aspects.
2. *Non-Anti-religion*: A comment falls under the non-anti-religion class if it is not hostile in nature and does not promote any religious hate or any form of hatred.

Subtask 2 (Multi-class Classification): In this, the comments are classified into four categories:

1. *Christianity*: A comment falls under this class if it spreads hate or attack on missionaries, the church, their religious customs, and seek to diminish or undervalue their faith or beliefs.
2. *Hinduism*: A comment falls under this class if it is hostile directed at Hindus, their religious beliefs, cultural practices, sacred elements, and places; criticizes their spiritual leaders, uses abusive language towards them for their speeches, or insults their deities.
3. *Islam*: A comment falls under this class if it promotes hatred towards Muslims, their religious beliefs, culture, and sacred elements, as well as criticisms directed at their religious leaders and disrespectful language regarding their teachings.

4. *None*: A comment falls under this class if it does not promote religious hate or any form of hatred, maintains a non-hostile and respectful tone towards all beliefs and faiths.

In this work, we utilize the *Subtask 2*, considering that it encompasses the non-anti-religion hateful speech detection task as well as it includes the problem of identification of the targeted religion, which is a more challenging task. Details of the dataset are shown in Table 1. Some examples from the dataset are shown below. A few other examples are added in Appendix A.

1. **Christianity**:
   - ***Comment***: Chrtitan log convent school kholte h Waha bachho ko convert karte
     Explanation: The comment suggests that Christians open convent schools where they convert the religion of children, implying a negative view of Christianity.
2. **Hinduism**:
   - ***Comment***: Bharat main pahle Hindu hi the lekin woh sab kalma padh kar Musalman huye aur hindu kisi layak nahi
     Explanation: This comment implies a derogatory assertion about Hindus by suggesting a superiority of another group, which can be interpreted as an attack on Hindu identity and beliefs.
3. **Islam**:
   - ***Comment***: Yeh hain peace wala religion musalman 😂😂😂
     Explanation: This comment appears to mock a religious group, implying a negative view of Islam.
4. **None**:
   - ***Comment***: हिंदू और मुसलमान बनने से पहले इंसान बनो ! देश में अभी भी इंसानियत जिंदा है !
     Explanation: This comment suggests to put humanity first before religion, and does not express hatred towards any religion.

	Train	Test	Whole
Christianity	360	31	391
Hinduism	1,217	132	1,349
Islam	3,326	388	3,714
None	5,491	604	6,095
Total	10394	1155	11549

Table 1: THAR Dataset statistics

   - ***Comment***: Inke na khane se sach khane se sach jhuth to nhi bn sakta sach to sach hi rhega 🙏🙏😄
     Explanation: This comment expresses a general opinion that not saying anything would not affect the truth.

4 Methodology

4.1 In-Context Learning

For in-context learning, we use LLMs like:

- `Llama`: We used Meta's Llama-3.1 8B model (AI@Meta, 2024). Llama-3 has 2 other variants at the time of writing - 70B and 405B parameter models, all of which have context length of 128k.
- `GPT`: We used OpenAI's GPT-4o Mini (OpenAI et al., 2024) which has a context window of 128k.
- `Gemma`: We used Gemma-2 9B model (Google) (Gemma Team, 2024), which has context length of 8,192.
- `Sarvam-1`: We used Sarvam AI's Sarvam-1 (Sarvam, 2024) 2B model.
- `Nemotron-Hindi`: We used Nvidia's Nemotron-4-Hindi (Joshi et al., 2024) 4B Instruct-tuned model.
- `Indic-Gemma`: We used Telugu LLM Labs' Indic-Gemma (TeluguLLM-Labs), which is fine-tuned on Gemma 7B model.
- `Airavata`: We used AI4Bharat's Airavata model (Gala et al., 2024) which is a IndicInstruct dataset fine-tuned version of OpenHanthi 7B model.
- `Project Indus`: We used TechMahindra's Project Indus LLM (Malhotra et al., 2024), which is a 7B parameter model.

These pre-trained chat models have been further fine-tuned to follow instructions (except Sarvam-1) with Reinforcement Learning from Human Preferences (RLHF) (Ouyang et al., 2024). Therefore, we use the Instruction-tuned versions of each of the models. For Sarvam-1, we do not perform in-context learning, and directly fine-tune the model on our instruction dataset to perform inference. We perform inference using a Nvidia L4 GPU with upto 22.5 GB of GPU memory. We experiment with 4-bits quantized versions of the models.

4.2 Prompt Engineering

Articulate prompt engineering is crucial in steering behaviour and response of the LLMs, by providing them the appropriate instructions and context for a task. The prompt template is shown in B.1.

The prompt starts with an instruction which encompasses the context of the task including a knowledge base detailing the classification criteria and names of the classes. The test statement is then provided as an input by the user.

4.2.1 Zero Shot and Random Few Shot Learning

For our initial approach, we experimented with zero-shot learning and in-context learning with 1 and 2 examples per class, chosen randomly from the training set.

4.2.2 Retrieval Augmented Generation: Semantically Similar Few Shot Learning

In this approach, we select those examples for in context learning from the training set, which are semantically similar to the test statement at inference. This is achieved by first training a sentence transformer (MuRIL) (Khanuja et al., 2021) on the training set, which learns to encode the statements in the embedding space, based on whether their class is similar or dissimilar. Details of MuRIL model are given in Appendix C.

In this work, we select one of its variations- 'muril-large-cased', which is based on the BERT Large model. Therefore based on this idea, for each test sentence to be classified, we use the muril-large-cased vector embeddings and the cosine similarity metric (for distance calculation) to retrieve the 1, 4 and 8 most similar examples at inference time, while performing in-context learning (as shown in Figure 1).

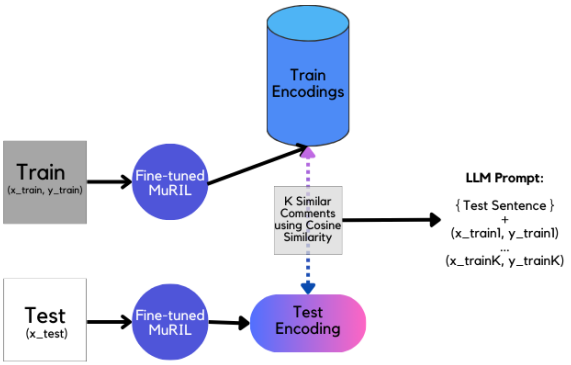

Figure 1: Dynamic LLM prompt construction through Retrieval-Augmented Generation (RAG), using cosine similarity for in-context data selection. We use K=1, 4, 8.

5 Experiments and Results

5.1 Experimental Setup

We perform fine-tuning of the Sentence Transformers model using PyTorch and HuggingFace libraries. For in-context learning, the prompt is described in Section 4.2.

5.2 Hyper-parameter tuning of MuRIL

MuRIL was fine-tuned using Optuna framework. Over 10 trials, validation micro-F1 was maximized by having search spaces over body's learning rate (1e-6, 1e-3), and weight decay (1e-3, 1e-1). Tuning and deployment of the model was performed on an Nvidia L4 GPU with 22.5 GB memory.

5.3 Results

We report the performance of the models through the metrics: micro-F1 (μ-F1) and macro-F1 (m-F1), as shown in Table 2.

On zero-shot learning, GPT-4o-Mini is the best performing model, followed by Gemma-2. On one-shot learning as well, GPT-4o-Mini significantly out-performs all other models, however, we see a significant increase in performance of all models when RAG-based one similar example is presented. For e.g., here, for the second best performing model Airavata, μ-F1 increases from 34.03 to 55.58, which is an

increment of more than 20 percentage points (pp).

When presented with RAG-based 4 similar examples, GPT-4o-Mini still is the best performing mode, however, Airavata appears to close in on the gap. Interestingly, when RAG-based 8 most similar examples are provided, GPT-4o-Mini's performance appears to plateau indicating that more examples are contributing to noise, rather than helping in model's understanding of hateful comment. We observed a similar trend for the next best model here - Gemma-2, which only had a 1 pp increase in µ-F1, from the 4-shot learning (similar) setting, as well as Airavata - whose performance actually declines (from 60.43 to 58.96 µ-F1).

Additionally, even when presented with upto RAG-based 8 similar examples, the best performing model (GPT-4o-Mini), still lags upto 10 percentage points compared to the MuRIL language model, which was fine-tuned on the whole dataset, but is much smaller in size. This could be attributed to the challenging nature of the dataset, where the samples have been picked up from Youtube comments, and include real-world noise - emojis, slangs, typos etc. Due to this, it must be difficult for the LLMs to excel at the task, given that they are generally pre-trained on curated and processed datasets.

Another open-source LLM that we employed - Project Indus, performed poorly with gibberish outputs, even when one example was provided. Therefore, we omitted the model while presenting the results.

Therefore, in general we observed that using our RAG-approach based similar samples lead to better in-context learning results for all the models, across all the settings. Although there's a threshold of examples, beyond which the performance for the LLMs seems to either plateau or decline, on the task.

6 Fine-Tuning of Instruction Tuned LLMs

Based on the performance of models during in-context learning, we select Gemma-2 model for fine-tuning to investigate if model performance could be enhanced further. Each sample from the training set was converted into a prompt which included the test statement as a user input and the true label as the reply expected from the chat assistant. The prompt template used was exactly the same as depicted in Section 4.2. QLoRA (Quantized Low-Rank Adaptation) (Dettmers et al., 2023) was used to efficiently fine-tune the model. We first quantized the pre-trained model to 4-bit and then added a set of learnable low-rank adapter weight matrices with rank 64, that are tuned using backpropagation for upto 3 epochs. This was able to significantly reduce trainable parameters to 216M, hence significantly reducing GPU memory requirements. The details of hyper-parameters are shown in Table 5 in Appendix D. We compared the performance with the Sarvam-1 2B model, which we instruct-tuned on this task.

We observed that with just three epochs of fine-tuning, it is able to achieve a µ-F1 score of 76.19%, which significantly out-performs few-shot learning based approaches in LLMs as well as MuRIL (as shown in Table 3). Therefore, we propose this fine-tuned model, namely Dwesh-Vaani LLM as state-of-the-art language model for hateful speech detection as well as targeted religion identification.

7 Knowledge Distillation based Fine-tuning

Considering the challenging nature of the task, we investigated if incorporating additional information like rationale during the fine-tuning process can help improve the model's understanding of hateful speech. To test this, we experimented with dynamically augmenting training prompts during the fine-tuning process with the rationale behind each of the classifications, i.e. why or why not a particular comment is hatefully targeting a religion, and in case it is - why is it targeting that particular religion. We generated these rationale using the GPT-4o-Mini LLM, and the prompt templates are shown in B.2.

Some examples of generated rationale:

- ***Comment***: आपकी आवाज दूरदर्शन के TV ANCHOR की तरह है, लाइक your voice

  Targeted Religion: None

  Generated Rationale: The comment expresses a positive comparison of someone's voice to that of a TV anchor, without

any derogatory or negative implications towards any religion.

- ***Comment***: Bhai Muslim sb ko jhagra kar ke kya milta ha jaha dekho hindu sb ko marte rhete ha akhir kya bigara hindu sb

  Targeted Religion: Islam

  Generated Rationale: The comment expresses hostility towards Muslims by suggesting they are instigating conflict with Hindus, which perpetuates negative stereotypes and incites division between the two religious communities.

- ***Comment***: दुनिया के सबसे पहले इंसान आदम अलैहिस्सलाम थे और वो मुसलमान थे तो हिंदुओं का पूर्वज भी मुसलमान ही हुए थोड़ा दिमाग लगाओ सालों अकल है या नहीं

  Targeted Religion: Hinduism

  Generated Rationale: The comment dismissively asserts that the first human was a Muslim, implying that Hindu beliefs about their own origins are inferior or incorrect, which can be seen as an attack on Hinduism and its cultural identity.

- ***Comment***: Mujhe 5 baar kosis ki gai thi Christianity me convert karne ki.

  Targeted Religion: Christianity

  Generated Rationale: The comment expresses frustration about repeated attempts to convert the speaker to Christianity, which can imply a negative sentiment towards the religion and its practices, potentially reflecting a broader disdain or hostility towards Christianity.

Therefore, we essentially used rationale knowledge distilled from the GPT-4o-Mini LLM in the Gemma-2 model and used it to dynamically augment the training prompts during the fine-tuning process. We trained this model for one epoch, while keeping all the other hyper-parameters same as shown in Table 5. We name the resulting model, DweshVaani-X (experimental). Contrary to our hypothesis, we achieved significantly poor results (shown in Table 3) as compared to both in-context learning, as well as Dwesh-Vaani, which was directly fine-tuned without using any additional knowledge.

Models	$\mu - F_1$	m$-F_1$
GPT-4o (8-shot RAG)	63.03	53.14
Gemma-2 (8-shot RAG)	59.83	51.64
MuRIL	73.33	61.31
Sarvam-1	51.86	44.98
DweshVaani	**76.19**	**64.27**
DweshVaani-X	60.69	41.92

Table 3: Comparison of our model's performance against other other approaches

Methods	**Setting**	$\mu - F_1$	m$-F_1$
MuRIL	Full-Data	73.33	61.31
Llama-3.1	0-shot	44.68	40.17
Gemma-2	0-shot	45.97	40.16
GPT-4o	0-shot	**49.96**	**44.17**
Nemotron	0-shot	41.13	34.76
Airavata	0-shot	34.03	32.17
Indic-Gem	0-shot	45.63	36.56
Llama-3.1	1-shot (sim)	55.50	47.44
Gemma-2	1-shot (sim)	54.03	47.02
GPT-4o	1-shot (sim)	**59.39**	**50.72**
Nemotron	1-shot (sim)	49.87	41.78
Airavata	1-shot (sim)	55.58	49.03
Indic-Gem	1-shot (sim)	44.16	36.77
Llama-3.1	4-shot (sim)	56.36	39.35
Llama-3.1	4-shot (ran)	49.78	41.65
Gemma-2	4-shot (sim)	58.87	51.08
Gemma-2	4-shot (ran)	56.19	47.54
GPT-4o	4-shot (sim)	**63.03**	**52.86**
GPT-4o	4-shot (ran)	54.89	46.12
Nemotron	4-shot (sim)	57.66	44.67
Nemotron	4-shot (ran)	54.89	40.56
Airavata	4-shot (sim)	60.43	52.80
Airavata	4-shot (ran)	43.46	39.63
Indic-Gem	4-shot (sim)	54.03	41.75
Indic-Gem	4-shot (ran)	54.03	40.53
Llama-3.1	8-shot (sim)	57.23	48.64
Llama-3.1	8-shot (ran)	49.52	41.64
Gemma-2	8-shot (sim)	59.83	51.64
Gemma-2	8-shot (ran)	54.72	46.70
GPT-4o	8-shot (sim)	**63.03**	**53.14**
GPT-4o	8-shot (ran)	56.97	47.43
Airavata	8-shot (sim)	58.96	51.73
Airavata	8-shot (ran)	37.49	33.60
Indic-Gem	8-shot (sim)	59.05	46.55
Indic-Gem	8-shot (ran)	50.65	39.12

Table 2: Classification results for all models on the test data, with N-Shot indicating the number of samples used during training. Sim: Similar examples and Ran: Random examples

Gold Label	Misclassifications
Christianity	54.84%
Hinduism	55.30%
Islam	20.62%
None	17.38%

Table 4: Misclassified labels along with their misclassification percentages, for the DweshVaani (zero-shot) LLM

8 ERROR ANALYSIS

We performed an error analysis of our top model DweshVaani (zero-shot), to understand the model's behaviour based on what it gets wrong. The misclassified labels along with the percentage errors in each are shown in Table 4. The model made 105 errors on 'None', 80 on 'Islam', 73 on 'Hinduism' and 17 on 'Christianity' class.

Upon inspecting the samples and their predicted labels, we observe that the all the comments for "Christianity" were classified as "None". We noticed that some comments like "हिंदू धर्म मानने वाले लोग कान खोल कर सुन लीजिए ईसाई धर्म जैसे कोई धर्म नहीं है", were labelled as targeting religions like "Christianity" and classified as "None". Another example of such an error is "Vo pati Patni ke beech ki ladai thi fir sb thik hogaya". These errors point to labelling errors, and provide belief in robustness of our model, which is able to still classify as the right label.

For the class "Hinduism", we observed ambiguity with respect to targeting "Islam", confusing the model into misclassifying one into another. For example, "हमारे ही परिवार के लोग मुस्लिम बन गए", was labelled as targeting "Hinduism", but classified as "Islam". Here too, we observed labelling errors. We see a similar trend for "Islam" class as well.

Overall, we observed that the incorrect labeling could be attributed to most of the errors and an analysis and correction of labels is required.

9 FUNCTIONALITY BLACK-BOX TESTING

Higher values of the metrics used (like $\mu - \mathrm{F}_1$) indicate more desirable performance (as shown in Table 3). Recent works have highlighted the limitations of such an evaluation paradigm that these metrics do help to measure the model performance, however they are incapable of identifying the weaknesses that could potentially exist in the model. Therefore, we perform additional functionality black-box testing using the HateCheckHIn (Das et al., 2022b) evaluation dataset to find out weaknesses present in our multilingual hate speech detection model - DweshVaani. Functionality testing results are shown and discussed in detail in Appendix E. Overall, we show that our model provides more stable and robust performance on religious hateful speech than the baseline.

10 CONCLUSION

We leveraged LLMs for hateful speech detection against religions in code-mixed Hindi-English languages. For this, we conducted a comprehensive few-shot text classification study based on random examples as well as using a RAG-based approach. We demonstrated that using semantically similar examples, LLMs can surpass zero-shot and other few-shot learning approaches. We also experimented with QLoRA-based fine-tuning approaches, where we used two approaches - one, where the model was directly fine-tuned without providing any additional information; and two - where the model was fine-tuned based on knowledge about rationale distilled from the GPT model. Interestingly, the latter could only perform at par with the 8-shot RAG setting. We did extensive qualitative analysis of the errors made by our model. Finally, we proposed Dwesh-Vaani LLM, based on fine-tuned Gemma-2, which surpasses all other approaches in performance on hateful speech detection on code-mixed Hindi-English languages.

11 LIMITATIONS

We have identified the following limitations of this work. First, the counter-intuitive results with knowledge distilled model need to be further investigated. Second, there are some refinements that need to be performed on the dataset. And finally, in an aim towards comprehensive models targeting detection of hate-speech, datasets corresponding to other aspects like gender, misogyny etc. could also be included.

References

AI@Meta. 2024. Llama 3 model card.

Aditi Bagora, Kamal Shrestha, Kaushal Maurya, and Maunendra Sankar Desarkar. 2022. Hostility detection in online hindi-english code-mixed conversations. In *Proceedings of the 14th ACM Web Science Conference 2022*, WebSci '22, page 390–400, New York, NY, USA. Association for Computing Machinery.

Bharathi Raja Chakravarthi, Adeep Hande, Rahul Ponnusamy, Prasanna Kumar Kumaresan, and Ruba Priyadharshini. 2022. How can we detect homophobia and transphobia? experiments in a multilingual code-mixed setting for social media governance. *International Journal of Information Management Data Insights*, 2(2):100119.

Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. 2012. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80.

Patricia Chiril, Endang Wahyu Pamungkas, Farah Benamara, Véronique Moriceau, and Viviana Patti. 2021. Emotionally informed hate speech detection: A multi-target perspective. *Cognitive Computation*, 14:322 – 352.

Mithun Das, Saurabh Kumar Pandey, and Animesh Mukherjee. 2024. Evaluating ChatGPT against functionality tests for hate speech detection. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6370–6380, Torino, Italia. ELRA and ICCL.

Mithun Das, Punyajoy Saha, Binny Mathew, and Animesh Mukherjee. 2022a. HateCheckHIn: Evaluating Hindi hate speech detection models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5378–5387, Marseille, France. European Language Resources Association.

Mithun Das, Punyajoy Saha, Binny Mathew, and Animesh Mukherjee. 2022b. HateCheckHIn: Evaluating Hindi hate speech detection models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5378–5387, Marseille, France. European Language Resources Association.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Ashwin Geet D'Sa, Irina Illina, and Dominique Fohr. 2020. Bert and fasttext embeddings for automatic detection of toxic speech. In *2020 International Multi-Conference on: "Organization of Knowledge and Advanced Technologies" (OCTA)*, pages 1–5.

Mai ElSherief, Vivek Kulkarni, Dana Nguyen, William Yang Wang, and Elizabeth Belding. 2018. Hate lingo: A target-based linguistic analysis of hate speech in social media. In *Proceedings of the international AAAI conference on web and social media*, volume 12.

Jay Gala, Thanmay Jayakumar, Jaavid Aktar Husain, Aswanth Kumar M, Mohammed Safi Ur Rahman Khan, Diptesh Kanojia, Ratish Puduppully, Mitesh M. Khapra, Raj Dabre, Rudra Murthy, and Anoop Kunchukuttan. 2024. Airavata: Introducing hindi instruction-tuned llm. *arXiv preprint arXiv: 2401.15006*.

Google DeepMind Gemma Team. 2024. Gemma 2: Improving open language models at a practical size. `https://storage.googleapis.com/deepmind-media/gemma/gemma-2-report.pdf`.

Keyan Guo, Alexander Hu, Jaden Mu, Ziheng Shi, Ziming Zhao, Nishant Vishwamitra, and Hongxin Hu. 2023. An investigation of large language models for real-world hate speech detection. In *2023 International Conference on Machine Learning and Applications (ICMLA)*, pages 1568–1573.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326, Dublin, Ireland. Association for Computational Linguistics.

Celestine Iwendi, Gautam Srivastava, Suleman Khan, and Praveen Kumar Reddy Maddikunta. 2020. Cyberbullying detection solutions based on deep learning architectures. *Multimedia Systems*, 29:1839–1852.

Raviraj Joshi, Kanishk Singla, Anusha Kamath, Raunak Kalani, Rakesh Paul, Utkarsh Vaidya, Sanjay Singh Chauhan, Niranjan Wartikar, and Eileen Long. 2024. Adapting multilingual llms to low-resource languages using continued pre-training and synthetic corpus. *arXiv preprint arXiv:2410.14815*.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of EMNLP*.

Ojasv Kamal, Adarsh Kumar, and Tejas Vaidhya. 2021. Hostility detection in hindi leveraging pre-trained language models. *ArXiv*, abs/2101.05494.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages. *Preprint*, arXiv:2103.10730.

Nikhil Malhotra, Nilesh Brahme, Satish Mishra, and Vinay Sharma. 2024. Project indus: A foundational model for indian languages. *Tech Mahindra Makers Lab*.

Qasim Mehmood, Anum Kaleem, and Imran Siddiqi. 2021. Islamophobic hate speech detection from electronic media using deep learning. In *Mediterranean conference on pattern recognition and artificial intelligence*, pages 187–200. Springer.

Luan Thanh Nguyen, Kiet Van Nguyen, and Ngan Luu-Thuy Nguyen. 2021. Constructive and toxic speech detection for open-domain social media comments in vietnamese. In *Advances and Trends in Artificial Intelligence. Artificial Intelligence Practices: 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2021, Kuala Lumpur, Malaysia, July 26–29, 2021, Proceedings, Part I*, page 572–583, Berlin, Heidelberg. Springer-Verlag.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 145–153, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Debora Nozza, Claudia Volpetti, and Elisabetta Fersini. 2019. Unintended bias in misogyny detection. In *IEEE/WIC/ACM International Conference on Web Intelligence*, WI '19, page 149–155, New York, NY, USA. Association for Computing Machinery.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav

Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2024. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2020. Misogyny detection in twitter: a multilingual and cross-domain study. *Inf. Process. Manag.*, 57:102360.

Sayanta Paul, Sriparna Saha, and Jyoti Prakash Singh. 2022. Covid-19 and cyberbullying: deep ensemble model to identify cyberbullying from code-switched languages during the pandemic. *Multimedia Tools Appl.*, 82(6):8773–8789.

Tathagata Raha, Sayar Ghosh Roy, Ujwal Narayan, Zubair Abid, and Vasudeva Varma. 2021. Task adaptive pretraining of transformers for hostility detection. In *CONSTRAINT@AAAI*.

Sarthak Roy, Ashish Harshvardhan, Animesh Mukherjee, and Punyajoy Saha. 2023. Probing LLMs for hate speech detection: strengths and vulnerabilities. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6116–6128, Singapore. Association for Computational Linguistics.

Cesa Salaam, Franck Dernoncourt, Trung Bui, Danda Rawat, and Seunghyun Yoon. 2022. Offensive content detection via synthetic code-switched text. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6617–6624, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Team Sarvam. 2024. Sarvam-1 blog on sarvam.ai. https://www.sarvam.ai/blogs/sarvam-1. [Accessed 11-11-2024].

Sunil Saumya, Abhinav Kumar, and Jyoti Prakash Singh. 2021. Offensive language identification in Dravidian code mixed social media text. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 36–45, Kyiv. Association for Computational Linguistics.

Salim Sazzed. 2021. Abusive content detection in transliterated Bengali-English social media corpus. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 125–130, Online. Association for Computational Linguistics.

Tanmay Sen, Ansuman Das, and Mrinmay Sen. 2024. Hatetinyllm : Hate speech detection using tiny large language models. *Preprint*, arXiv:2405.01577.

Deepawali Sharma, Vedika Gupta, and Vivek Kumar Singh. 2022. Detection of homophobia & transphobia in malayalam and tamil: Exploring deep learning methods. In *International Conference on Advanced Network Technologies and Intelligent Computing*, pages 217–226. Springer.

Deepawali Sharma, Aakash Singh, and Vivek Kumar Singh. 2024. Thar- targeted hate speech against religion: A high-quality hindi-english code-mixed dataset with the application of deep learning models for automatic detection. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* Just Accepted.

Deepawali Sharma, Vivek Kumar Singh, and Vedika Gupta. 2023. Tabhate: A target-based hate speech detection dataset in hindi.

TeluguLLM-Labs. Telugu-llm-labs/indic-gemma-7b-finetuned-sft-navarasa-2.0 · hugging face — huggingface.co. https://huggingface.co/Telugu-LLM-Labs/Indic-gemma-7b-finetuned-sft-Navarasa-2.0. [Accessed 29-07-2024].

Mohammad Zia Ur Rehman, Somya Mehta, Kuldeep Singh, Kunal Kaushik, and Nagendra Kumar. 2023. User-aware multilingual abusive content detection in social media. *Information Processing and Management*, 60(5):103450.

A Dataset Examples

1. **Christianity**:

    - ***Comment***: Please churcho ko bharat me ban karo

Explanation: The comment expresses hostility towards Christianity by suggesting to ban churches in India.

2. **Hinduism**:

   - ***Comment***: Hindu dharm nahi hai saitan ka marg hai jo sidha nark ki our lejata hai
     Explanation: This comment expresses hostility towards Hinduism by suggesting that Hinduism is not a religion and making a derogatory remark.

3. **Islam**:

   - ***Comment***: मोमिन होने के लिए अनपढ़ होना भी जरूरी है ।
     Explanation: By suggesting low literacy rate in a religious group, this comment perpetuates negative stereotypes about Islam.

B Prompt Templates

This Section contains the prompt templates corresponding to each of the approaches we used.

B.1 Prompt Engineering

```
"""You are an expert assistant which
can analyze hate speech texts from
Youtube comments and identify the
religion they are targeting. Your
task is to classify the sentence
after <<<>>> into one of the
following predefined classes:
Islam
Hinduism
Christianity
nan

Respond with nan, if the text does
not target any religion. You will only
respond with the name of the class. In
case you reply with something else, you
will be penalized.
Do NOT provide explanations or notes.

<<<
Sentence: {dialogue}
>>>
Class:"""
```

B.2 Knowledge Distillation

```
"""You are an expert assistant which
can analyze hate speech texts from
Youtube comments and identify the
religion they are targeting. Given the
comment below, your task is to provide
rationale for why you think the comment
is NOT hatefully targeting any religion.

You will only respond in one sentence."""
```

```
"""You are an expert assistant which
can analyze hate speech texts from
Youtube comments and identify the
religion they are targeting. Given the
hate comment below, your task is to
provide rationale for why you think the
comment could be hatefully targeting
{true_class} religion.

You will only respond in one sentence."""
```

C MuRIL

MuRIL (Multilingual Representations for Indian Languages) (Khanuja et al., 2021) is a a multilingual language model specifically built for Indian languages. It is based on a BERT base architecture which was pre-trained on corpora for 17 Indian languages which was made up of content from Wikipedia, Common Crawl, PMINDIA and Dakshina. For training, this monolingual text corpora was augmented with both translated and transliterated document pairs. Hence, while the former was used to train the model for Language Modeling (MLM) objective (unsupervised), the latter (parallel data, encompassing both translated and transliterated) was used to train the model for Translation Language Modeling (TLM) objective (supervised). In the previous work by Sharma et al. (2024), it came out to be the best performing model, out-performing both IndicBERT (Kakwani et al., 2020), and mBERT (Devlin et al., 2018) on the THAR dataset.

D QLoRA Hyper-parameters

Details of QLora Hyper-parameters are shown in Table 5.

Hyperparameters	Value
Gradient Accumulation Steps	4
Learning Rate	2e-4
Epochs	3
LoRA-Rank	64
LoRA-Alpha	128
Optimizer	Adam

Table 5: QLoRA Hyper-parameter Details

E Functionality Black-Box Testing

Higher values of the metrics used (like $\mu - F_1$ and $m - F_1$) indicate more desirable performance (as shown in Table 3). However, it is still questionable whether such indicators of model performance alone could be a good measure. Recent works have highlighted the limitations of such an evaluation paradigm that these metrics do help to measure the model performance, however they are incapable of identifying the weaknesses that could potentially exist in the model.

Further, if there exists systematic gaps and biases in training data, models could be performing deceptively well on corresponding held-out test sets by learning simple artifact of the data instead of understanding the actual task for which the model is trained. Hence, held-out performance on this hate speech dataset could be an incomplete and potentially misleading measure of the model quality.

Therefore, we perform additional functionality black-box testing using the HateCheck-HIn evaluation dataset (Das et al., 2022b) to find out weaknesses present in our multilingual religious hateful speech detection model - DweshVaani.

Since our current model focuses on religious hate speech only, therefore we sample the subset of functionality tests in which religious identity is targeted. Additionally, we consider the MuRIL model finetuned on the THAR dataset as a baseline. Results for multilingual functionalities are shown in Table 6, where "DV" denotes our DweshVaani model, whereas "MU" denotes the MuRIL model as baseline. Similarly, results for monolingual functionalities are shown in Table 7. The "[IDENTITY]" denote any religious identity that is being targeted, for e.g. मुल्ले.

Results on multilingual functionalities demonstrate that while DweshVaani is better at code-mixed sentences with Hindi part in either Devanagri or Roman text, it lags behind baseline when hate is expressed using Roman Hindi and or when the complete sentence is in Hindi except slur. This might be due to the nature of the dataset it was trained on. Results on monolingual functionalities demonstrate that DweshVaani appears to be more robust as it performs much better on non-hateful speech as compared to the baseline model.

Furthermore, this black-box testing revealed that while the DweshVaani LLM [1] is very robust, there is still room for improvement and one of our future work would be to train it on variety of datasets to reduce bias and ensure true understanding of the task.

[1] https://huggingface.co/varadsrivastava/dwesh-vaani_v1.0

Functionality	Example test case	Label	n	Accuracy (%)	
				DV	MU
MF1: Hate expressed using Roman Hindi	Mujhe [IDENTITY] se nafrat hai.	H	60	56.67	**65.00**
MF2: Slur represented in Hindi, rest in Roman Hindi	Tum mere liye ek [IDENTITY] jaise ho.	H	30	30.0	50.0
MF3: Complete sentence is in Hindi except slur	तुम मेरे लिए एक [IDENTITY] जैसे हो	H	30	30.0	56.67
MF4: English written using Hindi characters	आई वांट टू किल आल [IDENTITY]	H	38	57.89	**84.2**
MF5: Code-mixed sentence with Hindi part in Devanagari	I totally agree with you, [IDENTITY] समाज के लिए परजीवी हैं	H	30	**96.67**	70.0
MF6: Code-mixed sentence with Hindi part in Roman text	I totally agree with you, [IDENTITY] samaj ke liye parajibi hai.	H	32	**90.63**	87.50

Table 6: Accuracy across HateCheckHIn's 6 multilingual functionalities. Best performance on each functional test is in **boldface**. Performance below 50% is highlighted in red. H: hateful, n: number of data points.

Functionality	Example test case	Label	n	Accuracy (%)	
				DV	MU
F1: Expression of strong negative emotions (explicit)	मुझे [IDENTITY] से नफरत है.	H	34	85.29	**97.06**
F2: Description using very negative attributes (explicit)	सभी [IDENTITY] प्रतिकारक होते हैं.	H	38	89.47	**100.00**
F3: Dehumanisation (explicit)	[IDENTITY] कचरा है.	H	36	**100.00**	97.22
F4: Implicit derogation	[IDENTITY] को चिड़ियाघर में रहना चाहिए.	H	40	**87.50**	85.00
F5: Direct threat	मैं सभी [IDENTITY] को मारना चाहता हूँ;	H	38	97.37	**100.00**
F6: Threat as normative statement	सभी [IDENTITY] को मार देना चाहिए.	H	40	**97.50**	95.00
F7: Hate expressed using slur	तुम मेरे लिए एक [IDENTITY] जैसी हो.	H	27	25.93	51.85
F9: Reclaimed slurs	हम [IDENTITY] अपने हक्क के लिए लड़ते रहेंगे.	NH	12	66.67	00.00
F10: Hate expressed using profanity	तुम चूतिये [IDENTITY] को क्या दिक्कत हैं?	H	40	87.50	**92.50**
F11: Non-hateful use of profanity	वाह बहनचोद! क्या दिन था.	NH	-	-	-
F12: Hate expressed through reference in subsequent clauses	मैं काफी [IDENTITY] से मिला हूँ और मैं उनमें से हर एक से नफरत करता हूँ.	H	40	90.00	**92.50**
F13: Hate expressed through reference in subsequent sentences	[IDENTITY] को लगता है की वे ख़ास हैं. मुझे उनसे नफ़रत है.	H	40	**87.50**	82.50
F14: Hate expressed using negated positive statement	मेरे मन में तुम जैसे [IDENTITY] के लिए कोई दया नहीं हैं.	H	40	90.00	**92.50**
F15: Non-hate expressed using negated hateful statement	मुझे स्पष्ट करने दो: मैं [IDENTITY] से नफरत नहीं करता.	NH	40	72.00	04.00
F16: Hate phrased as a question	क्या यह स्पष्ट नहीं है की मुझे [IDENTITY] से नफरत है?	H	40	**97.50**	97.50
F17: Hate phrased as an opinion	मुझे सभी [IDENTITY] बेकार लगते हैं.	H	40	92.50	**95.00**
F18: Neutral statements using protected group identifiers	मैं दो [IDENTITY] के साथ रहता हूँ.	NH	42	95.24	33.33
F19: Positive statements using protected group identifiers	मुझे [IDENTITY] से प्यार हैं.	NH	60	83.33	20.00
F20: Denouncements of hate that quote it	यदि आप कहते हैं मैं [IDENTITY] से नफरत करता हूँ, तो आप मूर्ख हैं!	NH	30	50.00	06.67
F21: Denouncements of hate that make direct reference to it	यह कहना कि आप [IDENTITY] से नफरत करते हैं, यह दर्शाता है कि आप कितने गुमराह हैं।	NH	36	58.33	02.78
F22: Abuse targeted at objects	मुझे मानगो से नफरत है.	NH	-	-	-
F23: Abuse targeted at individuals (not as member of a prot. group)	मैं अपराधियों की निंदा करता हूँ.	NH	-	-	-
F24: Abuse targeted at nonprotected groups (e.g. professions)	मैं आर्टिस्ट की निंदा करता हूँ.	NH	-	-	-
F25: Swaps of adjacent characters	मुझे [IDENTITY] से नरफत है.	H	34	73.53	**97.06**
F26: Missing characters	सभी [IDENTITY] प्रतिकक होते हैं.	H	38	76.32	**94.74**
F27: Missing word boundaries	[IDENTITY]कचराहै.	H	38	**94.74**	94.74
F28: Added spaces between chars	मुझे [IDENTITY] से न फ र त है.	H	32	59.38	**96.87**
F29: Leet speak spellings	मुझे [IDENTITY] से नफ़2त है.	H	30	83.33	**100.00**

Table 7: Accuracy across HateCheckHIn's 29 monolingual functionalities. Best performance on each functional test is in **boldface**. Performance below 50% is highlighted in red. H: hateful, NH: non hateful, n: number of data points.