Hate Speech Detection in Somali-English Code-Switched Texts

Abdisalam Mahamed Badel[1], Ting Zhong[1], Xovee Xu[1], Wenxin Tai[1], and Fan Zhou[1,2](✉)

[1] University of Electronic Science and Technology of China, Chengdu 610054, China
{202214090105,xoee,wxtai}@std.uestc.edu.cn

[2] Kashi Institute of Electronics and Information Industry, Kashi, Xinjiang, China
{zhongting,fan.zhou}@uestc.edu.cn

Abstract. The use of large language models (LLMs) have grown significantly worldwide, offering numerous benefits but also posing risks of misuse. For example, LLMs can generate harmful content, such as hate speech targeting specific individuals or groups. Although recent research has begun addressing the detection of LLM-generated hate speech, low-resource languages remain markedly underrepresented. As LLMs are increasingly adopted by culturally and linguistically diverse communities, it is essential to evaluate their impact across all languages they are trained on, including those with limited resources. Code-switching, the practice of alternating between two or more languages within a single piece of content, presents unique challenges for automated hate speech detection. This study investigates the capability of LLMs to detect hate speech in Somali-English code-switched texts and introduces an evaluation framework that integrates local linguistic knowledge. We employ in-context few-shot learning and retrieval-augmented generation to enhance detection performance. Moreover, we develop a high-quality benchmark dataset consisting of 3,012 Somali-English code-switched texts containing explicit hate speech. Our findings reveal that while LLMs perform well in detecting hate speech in English segments, they struggle with the Somali segments, especially when the English portion expresses strong positive sentiment. Our proposed linguistic adjustments and strategies significantly enhance LLM performance in these multilingual and code-switched contexts. Our code and dataset are available at:https://github.com/Abdisalam-Badel/SECSHSD

Keywords: Hate speech · Somali · Large language model · Retrieval-augmented generation · Low-resource

Disclaimer: The content of this paper may include offensive language or text. We affirm that the offensive words mentioned do not represent our views.

X.-L. Mao et al. (Eds.): NLPCC 2025, LNAI 16104, pp. 364–376, 2026.
https://doi.org/10.1007/978-981-95-3349-7_28

1 Introduction

The rapid proliferation of social media applications in recent years has brought numerous advantages, including the facilitation of communication, information sharing, and access to resources. While these developments have simplified and enriched many aspects of daily life, they have also introduced significant challenges, some of which negatively affect users and may even lead to psychological trauma [24]. One of the most pressing concerns is the rise of hate speech, which has become increasingly pervasive on social media platforms [24]. Hate speech is broadly defined as offensive language directed at specific groups based on characteristics such as ethnicity, sexual orientation, gender, religion, nationality, or race [18]. Its widespread presence online poses serious challenges for both users and platform regulators [9]. In response, researchers have developed various methods to address this issue. For example, [15] introduced a benchmark dataset for explainable hate speech detection in English. More recent studies have employed pre-trained language models (PLMs), such as BERT and RoBERTa, which have been fine-tuned for hate speech detection tasks [14]. Furthermore, LLMs have emerged as promising tools in this domain. For instance, [21] utilized zero-shot learning with LLMs for hate speech detection. However, despite these advancements, significant gaps remain. Many low-resource languages, including Somali, are underrepresented in hate speech detection research, likely due to the scarcity of well-annotated datasets. Another major challenge arises when hate speech contains more than one language, a phenomenon known as code-switching, which refers to the alternation between two or more languages within a single comment or post [22]. This is particularly common in bilingual and multilingual communities. Although hate speech detection is a critical and growing area of research in the context of evolving social media, to the best of our knowledge, no prior studies have focused on Somali, particularly in code-switched scenarios. Therefore, the aim of this study is to investigate hate speech detection in Somali-English code-switched text, with a particular focus on evaluating the capabilities and safety of LLMs in this context. To this end, we constructed a dataset comprising 3,012 Somali-English code-switched samples, annotated into two categories: hate and normal. These samples were sourced from three major social media platforms: X (formerly Twitter), YouTube, and TikTok. We explore the effectiveness of LLMs in addressing this problem using in-context few-shot learning and retrieval-augmented generation (RAG). Additionally, we enhance model performance by incorporating local knowledge, such as hate-related Somali lexicons. Finally, our contributions are as follows: (1) We systematically evaluate the ability of LLMs to detect hate speech in Somali-English code-switched text, enhancing their performance through local knowledge of a hateful lexicon and RAG within a novel evaluation framework. (2) We introduce the first publicly available dataset specifically designed for Somali-English code-switched hate speech detection. (3) We conduct extensive experiments to explore various retrieval methods that improve hate speech detection in code-switched content.

2 Related Work

Previous research has explored various approaches to hate speech detection.

For example, [20] employed supervised machine learning techniques, while [13] utilized deep learning methods such as Bi-LSTM, LSTM, and CNN. Additionally, [8] proposed leveraging linguistic features to address hate speech in Spanish. Recently, research has increasingly focused on the adoption of PLMs. For instance, studies such as [14] employed models like mBERT and PTT5. More recently, LLMs, such as ChatGPT, have emerged as promising tools for hate speech detection. Notable contributions in this area include [9] and [19], which evaluated the performance of LLMs in this context. In addition to PLMs and LLMs, RAG techniques have gained attention for their potential in hate speech detection. For example, [26] explored the application of LLMs combined with RAG methods to detect abusive language, and [11] explored Multimodal method for Hate Detection. Further research has addressed hate speech detection across different languages and settings, including high-resource languages such as English [3], Chinese [5], and Hindi [2]. Low-resource languages, such as Wolof and Swahili [10]. Cross-lingual approaches, such as the one explored in [16], have further contributed to this area of research. Similarly, [22] examined hate speech detection in code-switched Hindi-English texts, where speakers use more than one language within a single post or comment. Code-switching introduces unique challenges for detection systems due to its linguistic complexity.

Despite these advancements, hate speech detection in many low-resource languages, including Somali, remains underexplored. This paper seeks to address this gap by investigating hate speech detection in Somali-English code-switched text.

3 Methodology

3.1 Overall Architecture

Figure 1 illustrates our workflow. We begin by processing the training data, dividing it into smaller chunks. Each chunk is transformed into embeddings using our embedding model and stored in a vector database (Vector DB). For the test data, each row is converted into an input, and relevant rows from the training data are retrieved for that specific input. Each input is transformed into embeddings and tokens, which are used to search for both semantically similar embeddings and lexically similar words within the vector database and the training data.

The results of the semantically related embeddings and lexically similar words are then combined. The top 3 most relevant rows for each input are retrieved to form the context. This context, along with the original input, is combined with a prompt and passed to the LLM. The response generated for each final prompt is then classified as either hate speech or normal text. It is important to note that, in this diagram, the training data serves as the external source.

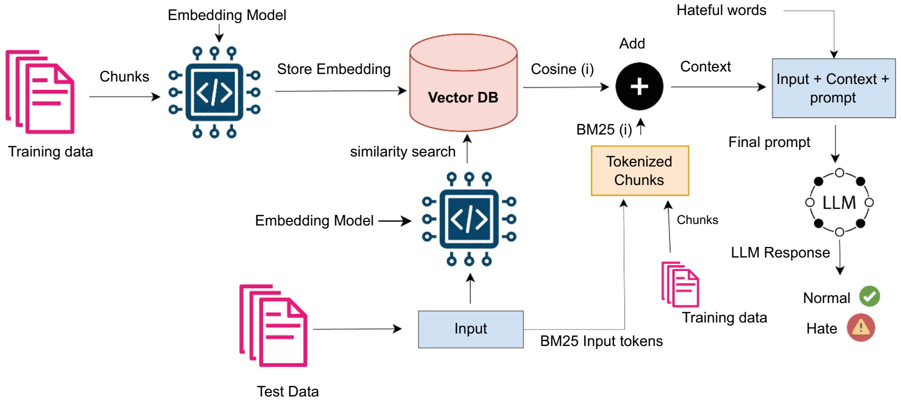


Fig. 1. Overall workflow: In this workflow, Cosine(i) is the normalized cosine similarity of the i^{th} row, while BM25(i) is the normalized BM25 of the i^{th} row. Here the Context contains top 3 relevant rows from training data to the input (query) from the test data.

We have divided the dataset into training and test sets with 1,812 and 1,200 samples, respectively. The reported findings in the paper are based on the test data. **Hateful words** refer to contextually hateful words and phrases. Our prompt configuration contains the **default parameters** for all the LLM models applied.

4 Experiments

4.1 Dataset

We constructed a new dataset to address the detection of Somali-English code-switched hate speech. The dataset consists of user-generated comments sourced from three prominent social media platforms: X, YouTube, and TikTok. The data statistics include a total of 3,012 samples, of which 1,862 are labeled as ☹ (hate speech) and 1,150 as ☺ (normal). The annotation process was carried out by three native Somali speakers, who engaged in iterative discussions to ensure label quality and consistency. All annotators possess a strong command of English, enabling them to accurately interpret and label content in both languages.

The dataset has been anonymized to protect user privacy and contains no personally identifiable information. Furthermore, we adhered to the data collection policies of X, YouTube, and TikTok, ensuring that only publicly available user-generated text was collected. The overall annotation process was supervised by the first author, a native Somali speaker, to maintain accuracy and reliability.

Table 1 presents several examples from the benchmark dataset, accompanied by their full English translations. In these translations, *Daarood* refers to a Somali clan, while *iidoor* is a derogatory term used as an insult against another Somali clan. *Abaarso* is the name of a school, but in this context, it is mentioned in a derogatory manner. The term is used as an insult, implying that being a graduate of this school reflects poorly on an individual's character or culture, thereby portraying the school in a negative light. These words, along with many others, are frequently used in written formats or live podcasts during clan-based debates. Recent reports from Somalia indicate that hate speech in the country, recovering from clan-based civil wars waged by clan militias, may worsen the situation [1]. In this study, we classify offensive language, abusive text, cyberbullying, sexism, and similar forms of harmful expression under the umbrella term hate speech. Please note that, throughout the paper, we use the terms code-switched and code-mixed interchangeably. In conclusion three annotators participated in the task. We determined disagreements through a voting process applied to the samples we annotated. In the first round, we got a sample-level annotation agreement of over 73%. However, we conducted a discussion process covering three rounds to achieve agreement on all samples and resolve any inconsistencies. The dataset is entirely code-switched, meaning every sample in the dataset contains code-switches, making it a 100% code-switched dataset. Finally, we informed our annotators about the planned use of the dataset and paid them fairly. We also state that we created this dataset for academic research purposes.

Table 1. Sample annotations from the dataset.

Text	Label
(1). Daarood dabo ku dhiigle, we not ready to listen you **Translation**: Daarood the blood ass, we are not ready to listen you	Hate
(2). For me, it is ok, *uma arko wax dhibleh* **Translation**: For me, it is okay; I do not see it as a problem.	normal
(3).*Dhaqaniyan* they are known as thieves, *mooryaan tuugo ah* **Translation**: Culturally, they are known for theft and stealing.	Hate
(4). Cry as usual *iidoor abaarso, hooyadiin wasee* **Translation**: Cry as usual, *iidoor, abaarso*, f*ck your mothers.	Hate

4.2 Implementation Details

Using the Transformers library [25], we employed the **sentence-transformers/all-MiniLM-L6-v2** model to generate sentence embeddings. Retrieval was performed through a hybrid approach that combined **BM25** [23] with cosine similarity. BM25 was selected for its effectiveness in capturing lexical similarity and ranking, particularly in longer documents, while cosine similarity was used to capture semantic relationships between texts. We evaluated three ChatGPT variants: **gpt-3.5-turbo**, **gpt-4**, and **gpt-4-mini** [17], alongside the **DeepSeek-V3** [4] model. To improve detection accuracy and efficiency, we adopted a 1-shot in-context learning prompting strategy. In our hybrid retrieval system, the top-k relevant rows were selected by normalizing retrieval scores and averaging the outputs of BM25 and cosine similarity. This ensured that the top-k retrieved rows (set to 3 in our case) served as contextual input for subsequent processing.

Notably, BM25 operates directly on tokenized query terms without requiring embedding generation, as described in our workflow. The top-k rows were identified based on Eq. 1, while cosine similarity was computed using Eq. 2, which follows the commonly used dot product formulation.

Evaluation Metrics: To evaluate performance, we employ several metrics, including Recall, Precision, Accuracy, F_1-score, and Macro-F_1.

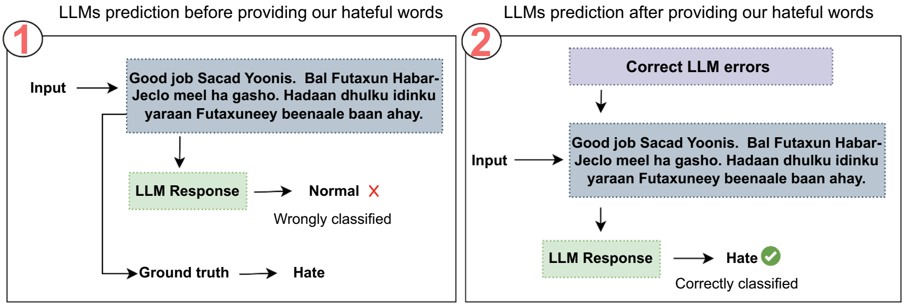

LLMs prediction before providing our hateful words
1
Input
Good job Sacad Yoonis. Bal Futaxun Habar-Jeclo meel ha gasho. Hadaan dhulku idinku yaraan Futaxuneey beenaale baan ahay.
LLM Response
Normal ✗
Wrongly classified
Ground truth
Hate
LLMs prediction after providing our hateful words
2
Correct LLM errors
Input
Good job Sacad Yoonis. Bal Futaxun Habar-Jeclo meel ha gasho. Hadaan dhulku idinku yaraan Futaxuneey beenaale baan ahay.
LLM Response
Hate ✓
Correctly classified

Fig. 2. Demonstrates how we enhanced the LLMs' ability to understand Somali text. The left side presents its output prior to the addition of hateful words, while the right side illustrates its improved performance after the introduction of this hateful words to the LLMs.

$$\text{topK}(i) = \alpha \cdot \text{Cosine}(i) + (1 - \alpha) \cdot \text{BM25}(i) \tag{1}$$

We set $\alpha = 0.5$, where α is the weighting factor that balances our retrievers, i denotes the i^{th} row in the dataset, Cosine(i) is the normalized cosine similarity of the i^{th} row, and BM25(i) is the normalized BM25 score of the i^{th} row.

$$\text{similarity}(\theta) = \frac{a \cdot b}{\|a\| \times \|b\|} \tag{2}$$

where a is the embeddings of the input (query) and b is the embeddings of training data (rows).

4.3 Baseline Models

We adopted three baseline systems: (1) LLMs Only: We experimented with LLMs without additional retrieval. (2) Dense Retrieval with FAISS: We used FAISS [7], a vector database and indexing-based library developed by Meta (Facebook), for dense retrieval. In this setup, FAISS was used for indexing and vector database management, while embeddings were generated using the `sentence-transformers/all-MiniLM-L6-v2` model. (3) Fine-Tuned PLMs: We utilized two PLMs, BERT [6] and RoBERTa [12], both of which are well established for classification tasks. These models were fine-tuned over three epochs with a learning rate of 2e-5 and a batch size of 32.

5 Main Results

Our hybrid approach, as presented in Table 2, significantly outperforms the baseline systems, with DeepSeek-V3 and gpt-3.5-turbo achieving accuracies of 78.33% and 77.17%, respectively. A detailed performance breakdown is also provided in Table 2, and Table 3, while the performance of DeepSeek-V3 is visualized in Fig. 5a. Despite these improvements, misclassifications remain, particularly when English portions are strongly positive, as demonstrated in the resolved example in Fig. 2. This highlights the persistent challenges of adapting models to low-resource languages. FAISS showed the lowest performance, likely due to its reliance on large corpora, an inherent limitation in Somali-language contexts, given our limited resources and small dataset. Figure 5b illustrates correlation percentages for specific slurs identified in the dataset.

Performance Analysis of Baseline Systems. As shown in Table 3, LLM-only approaches underperform compared to RAG-augmented models, although DeepSeek-V3 performs very closely to our hybrid method. For example, RoBERTa and BERT achieve lower accuracy than their hybrid counterparts (see Table 2), underscoring the advantages of incorporating local linguistic knowledge, such as contextually hateful words. While LLM-only systems demonstrate strong performance in individual target categories as can be seen Fig. 3b, their Macro-F_1 scores remain lower than those of RAG-enhanced LLMs, as illustrated in Fig. 4a.

Table 2. Performance comparison of our hybrid approach with baseline systems.

Retriever	Model	Precision		Recall		F_1-score		Accuracy
		Hate	Normal	Hate	Normal	Hate	Normal	
–	BERT	64.04	54.05	88.44	21.51	74.29	30.77	62.50
–	RoBERTa	61.97	66.67	98.64	4.30	76.12	8.08	62.08
FAISS	gpt-3.5-turbo	33.33	38.53	0.27	99.14	0.54	55.49	38.50
	gpt-4o-mini	1.00	38.67	–	1.00	–	55.77	38.67
	gpt-4o	1.00	38.67	–	1.00	–	55.77	38.67
	deepseek-V3	–	38.62	–	99.78	1.00	55.68	38.58
Hybrid	gpt-3.5-turbo	78.17	75.00	87.09	61.42	82.39	67.54	77.17
	gpt-4o-mini	89.05	61.96	66.30	87.07	76.01	72.40	74.33
	gpt-4o	96.14	53.76	47.42	96.98	63.51	69.18	66.58
	deepseek-V3	84.29	70.16	79.48	76.51	81.82	**73.20**	**78.33**

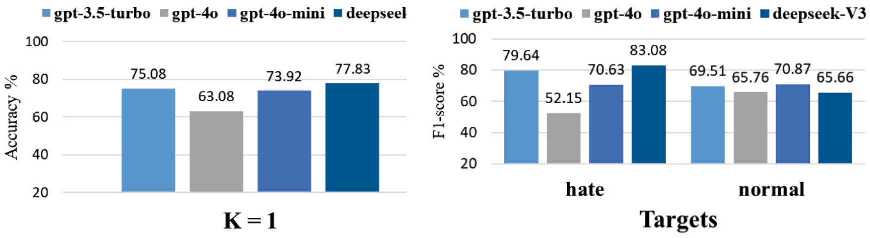
These findings suggest that LLM performance can be significantly improved when supplemented with language-specific data.

Table 3. Performance comparison of our hybrid approach with LLMs-only (Retr = Retriever, Acc = Accuracy).

Retr	Model	Hybrid					LLMs only				
		Precision		Recall		Acc.	Precision		Recall		Acc.
		hate	normal	hate	normal		hate	normal	hate	normal	
Hybrid	gpt-3.5-turbo	78.17	75.00	87.09	61.42	77.17	81.51	67.20	77.85	71.98	75.58
	gpt-4o-mini	89.05	61.96	66.30	87.07	74.33	91.94	57.62	57.34	92.03	70.75
	gpt-4o	96.14	53.76	47.42	96.98	66.58	98.49	49.20	35.46	99.14	60.08
	deepseek-V3	84.29	70.16	79.48	76.51	**78.33**	76.61	79.27	90.76	56.03	77.33

5.1 Ablation Study

We conducted two ablation studies: (1) examining the effects of retrieved relevant rows by setting k=1, and (2) assessing the effectiveness of retrieval methods by testing BM25 and cosine similarity individually. Figure 4b shows a Macro-F_1 comparison of the different retrievers, while Table 4 presents their overall results. Both BM25 and cosine similarity retrievers perform well, as shown in Table 4, emphasizing the importance of robust retrieval in code-switched hate speech detection. However, the benefits of RAG vary across models: the performance gain for GPT-4o exceeds that of DeepSeek-V3 during with and without RAG, indicating potential model-specific adaptation requirements. Due to **budget constraints**, our ablation analysis was limited to $k = 1$ retrieval results, as shown in Fig. 3a, where DeepSeek-V3 achieved the highest accuracy. Notably, zero-shot analysis was excluded from this evaluation.



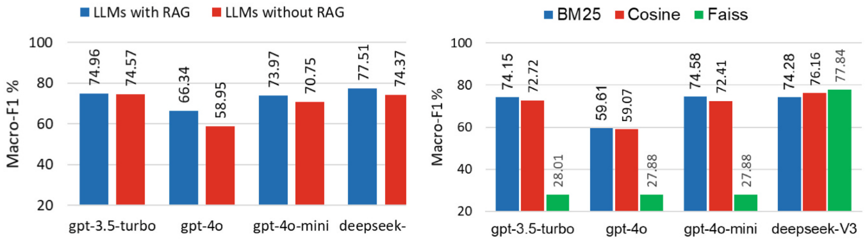
(a)

(b)

Fig. 3. Accuracy of our hybrid approach when K = 1 (a) and F_1-score comparison of individual hate and normal targets in LLMs-only (b).

Table 4. Performance of individual BM25 and cosine similarity retrieval methods (Retr = Retriever).

Retr	Model	Precision		Recall		F_1-score		Accuracy
		Hate	Normal	Hate	Normal	Hate	Normal	
BM25	gpt-3.5-turbo	77.66	73.75	86.41	60.56	81.80	66.51	76.42
	gpt-4o-mini	88.45	62.64	67.66	85.99	76.67	72.48	74.75
	gpt-4o	98.89	49.57	36.28	99.35	53.08	66.14	60.67
	deepseek-V3	75.74	84.91	94.16	52.16	83.95	64.62	77.92
Cosine	gpt-3.5-turbo	84.25	61.81	69.02	79.53	75.88	69.56	73.08
	gpt-4o-mini	91.75	59.30	60.46	91.38	72.89	71.93	72.42
	gpt-4o	98.13	49.25	35.73	98.92	52.39	65.76	60.17
	deepseek-V3	79.19	76.03	87.36	63.58	83.07	69.25	**78.17**



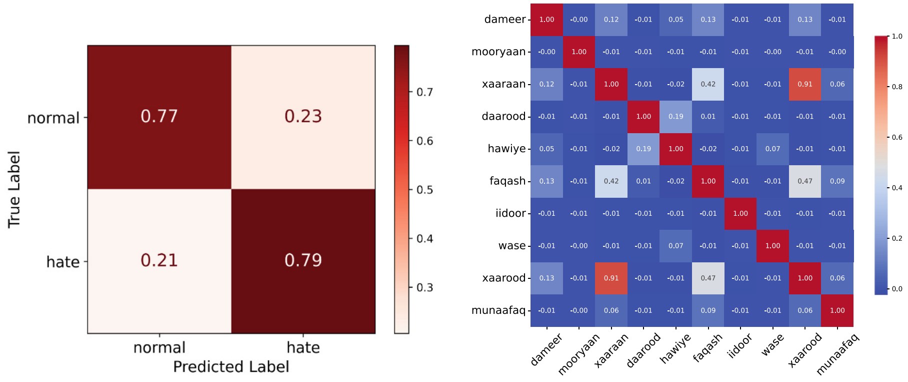
(a) (b)

Fig. 4. Macro-F_1 comparison of our hybrid approach with LLMs-only (a) and different retrievers (b).

5.2 Error Analysis

As discussed in Sect. 5 and illustrated in Fig. 2, the proposed model effectively identifies the English portions of the code-switched data. However, the model fails to detect *futaxun* as hate speech targeting a group. This failure occurs because the model's understanding is influenced by the presence of the English phrase *good job* at the beginning of the sentence. Similarly, in another sample, the word *fuleeynimada* is mistakenly perceived as positive text, even though it constitutes a hate speech. These examples highlight that positive words in the sentences can weaken our model's capacity to detect hate words. Lastly, although various words were misclassified, these words are those we recorded in the first run of our experiments.

Takeaways. Our study finds that while LLMs effectively detect hate speech in high-resource languages like English, their performance declines significantly in low-resource languages, such as Somali. This gap underscores the limitations of LLMs in protecting users from linguistically marginalized communities.



(a)

(b)

Fig. 5. Confusion matrix of DeepSeek-V3 on our hybrid approach (a) and (b) correlation percentage of selected hate slurs in the dataset.

However, our findings suggest that leveraging high-quality annotated datasets for low-resource languages is crucial in addressing these shortcomings and improving model robustness. Evidence for this is presented in Fig. 2, where we demonstrate how correcting the LLM enabled it to accurately classify a text that was initially misclassified.

6 Conclusion

In this paper, we propose a novel framework for detecting code-switched Somali-English hate speech. Our primary objective is to enhance the safety and reliability of LLMs when processing code-switched Somali-English text, a linguistic phenomenon common among Somali speakers. We conducted the study in two phases. First, we constructed a benchmark dataset comprising 3,012 Somali-English text samples. Second, we developed an evaluation framework that integrates LLMs with RAG and contextually hateful words to identify hate speech.

Our findings indicate that while LLMs perform well in detecting English hate speech, they encounter significant challenges when processing Somali. Nevertheless, we demonstrate that LLM performance in low-resource languages can be substantially improved through the use of high-quality, language-specific datasets. Overall, our contributions, including the benchmark dataset, offer a valuable resource for researchers and organizations aiming to build safer and more inclusive language technologies for linguistically diverse communities. However, this study represents a significant starting point. We encourage further research to build upon our findings and address the remaining challenges in detecting hate speech in low-resource settings.

Limitations. Our study has several limitations that present opportunities for future research. First, the limited size of our dataset may require future expansion. Second, budget constraints restricted our ablation study to a K value of 1. This restriction stems from the significant cost increases associated with higher K values. For example, when K = 2, the dataset expands to three times its original size (2 + 1 = 3 samples per prompt), which increases the number of tokens per prompt and, consequently, the financial cost. Future studies could investigate the effects of higher K values to achieve more comprehensive analyses. Third, our study focused on a few-shot approach and did not include a zero-shot analysis, primarily due to budgetary limitations. Finally, we encourage the research community to expand the dataset to address these constraints. Moreover, we recommend that AI companies developing LLMs prioritize mitigating the challenges associated with these models, especially in scenarios where the English portion of code-switched data conveys strong positive sentiment.

Open Science

We are prepared to share our developed artifacts, including the code, designed prompts, dataset, and setup scripts, with the scientific community. This may be particularly beneficial for research on low-resource languages.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China under Grant 62072077 and Grant 62176043; in part by the Kashgar Science and Technology Bureau under Grant KS2023025.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Hiiraan Homepage,https://hiiraan.com/news4/2024/Nov/198966/tiktok_clan_battles_over_music_and_poetry_stoking_tensions_in_somalia.aspx, Accessed 05 Jan 2025
2. Das, M., Saha, P., Mathew, B., Mukherjee, A.: HateCheckHIn: evaluating hindi hate speech detection models. In: Calzolari, N., et al., (eds.) Proceedings of the Thirteenth Language Resources and Evaluation Conference, pp. 5378–5387. European Language Resources Association, Marseille, France (2022). https://aclanthology.org/2022.lrec-1.575/
3. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language **11**, 512–515 (2017). https://doi.org/10.1609/icwsm.v11i1.14955
4. DeepSeek, Inc.: Your first api call — deepseek api docs. https://api-docs.deepseek.com/ (2025), Accessed 05 Apr 2025
5. Deng, J., Zhou, J., Sun, H., Zheng, C., Mi, F., Meng, H., Huang, M.: COLD: A benchmark for Chinese offensive language detection. In: Goldberg, Y., Kozareva, Z., Zhang, Y. (eds.) Proceedings of the 2022 Conference on Empirical Methods

in Natural Language Processing, pp. 11580–11599. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (2022). https://aclanthology.org/2022.emnlp-main.796/

6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1423
7. Douze, M., et al.: The faiss library (2025). https://arxiv.org/abs/2401.08281
8. García-Díaz, J.A., Jiménez-Zafra, S.M., García-Cumbreras, M.A., Valencia-García, R.: Evaluating feature combination strategies for hate-speech detection in Spanish using linguistic features and transformers. Complex Intell. Syst. **9**(3), 2893–2914 (2023). https://doi.org/10.1007/s40747-022-00693-x
9. Guo, K., et al.: An investigation of large language models for real-world hate speech detection (2024). https://arxiv.org/abs/2401.03346
10. Jacobs, C., Rakotonirina, N.C., Chimoto, E.A., Bassett, B.A., Kamper, H.: Towards hate speech detection in low-resource languages: comparing asr to acoustic word embeddings on wolof and swahili (2023). https://arxiv.org/abs/2306.00410
11. Lang, J., Hong, R., Xu, J., Li, Y., Xu, X., Zhou, F.: Biting off more than you can detect: retrieval-augmented multimodal experts for short video hate detection. In: Proceedings of the ACM on Web Conference 2025, pp. 2763–2774. WWW '25, Association for Computing Machinery, New York (2025). https://doi.org/10.1145/3696410.3714560
12. Liu, Y., et al.: Roberta: a robustly optimized bert pretraining approach (2019). https://arxiv.org/abs/1907.11692
13. Malik, J.S., Qiao, H., Pang, G., van den Hengel, A.: Deep learning for hate speech detection: a comparative study. Int. J. Data Sci. Anal. (2024). https://doi.org/10.1007/s41060-024-00650-6
14. Masud, S., Khan, M.A., Goyal, V., Akhtar, M.S., Chakraborty, T.: Probing critical learning dynamics of PLMs for hate speech detection. In: Graham, Y., Purver, M. (eds.) Findings of the Association for Computational Linguistics: EACL 2024, pp. 826–845. Association for Computational Linguistics, St. Julian's, Malta (2024). https://aclanthology.org/2024.findings-eacl.55/
15. Mathew, B., Saha, P., Yimam, S.M., Biemann, C., Goyal, P., Mukherjee, A.: Hatexplain: a benchmark dataset for explainable hate speech detection (2022). https://arxiv.org/abs/2012.10289
16. Nozza, D.: Exposing the limits of zero-shot cross-lingual hate speech detection. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (vol. 2: Short Papers), pp. 907–914. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.acl-short.114
17. OpenAI: openai API. https://openai.com/api/ (2024), Accessed 02 Jan 2025
18. Pan, R., Antonio García-Díaz, J., Valencia-García, R.: Comparing fine-tuning, zero and few-shot strategies with large language models in hate speech detection in english. CMES - Computer Modeling in Engineering and Sciences **140**(3), 2849–2868 (2024). https://doi.org/10.32604/cmes.2024.049631, https://www.sciencedirect.com/science/article/pii/S1526149224000493

19. Podolak, J., Łukasik, S., Balawender, P., Ossowski, J., Piotrowski, J., Bakowicz, K., Sankowski, P.: LLM generated responses to mitigate the impact of hate speech. In: Al-Onaizan, Y., Bansal, M., Chen, Y.N. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2024, pp. 15860–15876. Association for Computational Linguistics, Miami (2024). https://doi.org/10.18653/v1/2024.findings-emnlp.931
20. Pyingkodi, M., et al.: Hate speech analysis using supervised machine learning techniques. In: 2023 International Conference on Computer Communication and Informatics (ICCCI), pp. 1–6 (2023). https://doi.org/10.1109/ICCCI56745.2023.10128591
21. Roy, S., Harshvardhan, A., Mukherjee, A., Saha, P.: Probing LLMs for hate speech detection: strengths and vulnerabilities. In: Bouamor, H., Pino, J., Bali, K. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 6116–6128. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.findings-emnlp.407
22. Sreelakshmi, K., Premjith, B., Soman, K.: Detection of hate speech text in hindi-english code-mixed data. Procedia Comput. Sci. **171**, 737–744 (2020). https://doi.org/10.1016/j.procs.2020.04.080, https://www.sciencedirect.com/science/article/pii/S1877050920310498, third International Conference on Computing and Network Communications (CoCoNet'19)
23. Trotman, A., Puurula, A., Burgess, B.: Improvements to bm25 and language models examined. In: Proceedings of the 19th Australasian Document Computing Symposium, pp. 58–65. ADCS '14, Association for Computing Machinery, New York (2014). https://doi.org/10.1145/2682862.2682863
24. Watanabe, H., Bouazizi, M., Ohtsuki, T.: Hate speech on twitter: a pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. IEEE Access **6**, 13825–13835 (2018). https://doi.org/10.1109/ACCESS.2018.2806394
25. Wolf, T., et al.: Transformers: state-of-the-art natural language processing. In: Liu, Q., Schlangen, D. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.emnlp-demos.6
26. Yao, T., Foo, E., Binnewies, S.: Personalised abusive language detection using LLMs and retrieval-augmented generation. In: Abbas, M., Freihat, A.A. (eds.) Proceedings of the 7th International Conference on Natural Language and Speech Processing (ICNLSP 2024), pp. 92–98. Association for Computational Linguistics, Trento (2024). https://aclanthology.org/2024.icnlsp-1.11/