LLM-based conversational systems for Sinhala: Investigating model limitations and exploring a framework for agentic retrieval

Malithi Alahapperuma
Department of Information Studies
University College London
malithi.alahapperuma.20@ucl.ac.uk

Andreas Vlachidis
Department of Information Studies
University College London
a.vlachidis@ucl.ac.uk

Antonis Bikakis
Department of Information Studies
University College London
a.bikakis@ucl.ac.uk

Abstract

Large Language Models (LLMs) have introduced significant shifts to the way we communicate and seek information. However, LLMs perform suboptimally in low-resource languages. Given their unprecedented capabilities, LLMs provide a unique opportunity for under-served linguistic communities to access knowledge and skills that were hitherto inaccessible. Therefore, finding practical and realistic ways for low-resource language users to access LLM-based conversational systems can have far-reaching societal and economic benefits. In this study, we characterise three limitations that reduce the usability of LLM-based conversational systems for Sinhala speakers and conceptually explore the design of an LLM agent-based retrieval framework.

1 Introduction

The introduction of Large Language Models (LLMs) for wider public use has created a paradigm shift in the way we communicate and seek information. However, LLMs perform poorly in low-resource languages (Kshetri, 2024; Zhong et al., 2024). A language is considered low-resource when it lacks data, artefacts, resources, and institutional support (Nigatu et al., 2024). Due to their immense language processing capacity, LLMs provide a hitherto unavailable opportunity to support low-resource languages. Providing equitable access to language technology is an opportunity to correct the historical inequalities faced by under-represented linguistic communities across the globe (Smart et al., 2024). Against this backdrop, this study aims to bring together advances in LLM workflows to improve LLM-based conversational systems for Sinhala. We specifically focus on workflows that can be practically deployed to non-expert users in low-resource contexts. The work discussed in this paper is exploratory and is intended to provide a conceptual foundation upon which to develop further system specifications.

2 Setting the landscape

There are several threads of work in progress to improve the performance of LLMs for low-resource languages. Two key approaches are model fine-tuning and developing LLM-based workflows. In fine-tuning, LLMs are further trained on language data from a given low-resource language. In the language context of this study, Sinhala, we come across SinLlama, an LLM pretrained and fine-tuned for three tasks, writing-style classification, news categorisation and sentiment analysis (Aravinda et al., 2025). The authors of the study show how this targeted fine-tuning significantly improves the performance of the LLM for the selected tasks (Aravinda et al., 2025). Similar approaches to language-specific adaptations of language models are seen across other low-resource language contexts (Joshi et al., 2024; Tonja et al., 2024).

However, improving underlying model performance does not entirely solve the issues encountered in low-resource contexts. In order to address these problems, LLM-based workflows with additional features and functionalities have been explored. Retrieval Augmented Generation (RAG) was conceptualised to provide non-parametric memory (external data sources) to LLMs, with the goal of improving the retrieval performance of LLM-based systems (Lewis et al., 2020). RAG has been used in low-resource contexts to improve LLM-based systems. Ndimbo et al. (2025) have built a RAG-based conversation system for Swahili, and Christian et al. (2025) have built an adaptive

Proceedings of the International Conference LAnguage TEchnologies for Low-resource Languages, pages 304–308
28 – 30 September 2026, Fes, Morocco ©LaTeLL 2026
https://doi.org/10.26615/978-954-452-112-7_031

RAG system for Indonesian. Alshammary et al. (2025) present a RAG framework for Arabic named ARCADIA (Arabic Retrieval Contextual Analysis Data-Intensive Applications). Alshammary et al. (2025) note that capturing complex nuances and the rich morphology of Arabic is a particular challenge in working with Arabic, which we particularly note as relevant to our study.

Given the focus of our work on developing LLM-based conversational systems that can be practically deployed in the Sinhala-speaking context, the feasibility of model fine-tuning as a solution is limited. This is particularly due to constraints in data and financial and technical resources. Given the scalability, adaptability, and reusability of modular LLM-based workflows, we believe a system that brings together RAG and agentic capabilities is best suited for further exploration.

3 Investigating model behaviour

Prior to conceptualising the system, we use a set of experiments to investigate the specifics of the challenges encountered when using LLM-based conversational systems in Sinhala. The experiments focus on three tasks: (1) text generation, (2) translation, and (3) information retrieval. These tasks are specifically selected given their relevance for practical use by non-expert users. For similar reasons, the queries used in the experiments are framed to represent the kind of questions a non-expert user would ask an LLM during a daily interaction. For each query, we generate responses using three widely available LLM conversational tools: ChatGPT (based on GPT 5.5), Claude (based on Sonnet 5 Medium) and DeepSeek (based on DeepSeek-V4). ChatGPT and Claude were selected because they are commonly used tools. DeepSeek was included to examine whether an open-weights multilingual model would provide different insights. Close reading of the generated texts is used to draw initial observations and trends. We do not use any computational analysis at this point, but rather aim to uncover broad themes in the limitations observed. The findings of the experiments are discussed below under each task category. Altogether, we used five queries under each task, with three responses generated per query from the different models. In order to best utilise the space available, we only provide a few salient examples in the discussion. All LLM output, prompts used, and further discussion are available in the project GitHub repository, available at: https://github.com/malithia/diagnostic-bench-llms-sinhala/.

Experiment 01: Text generation

In this task, we asked the models to generate texts for various scenarios. We select two examples to compare here: (1) A cover letter for a job application for the role of a driver, and (2) a Facebook post to advertise a clothing business. In evaluating the responses, we note that for the cover letter, which is a formal piece of writing, the LLMs performed relatively well. The tone of the text was appropriate and the content of the letter was accurate and suitable. There were no glaring errors in grammar and syntax, but, with some variability across models, the choice of words was sometimes unsuitable. However, the output of the Facebook post was noticeably poor across all three models. The responses contained grammatical and syntactic errors. The most prominent drawback was the tone, which was a mix of formal and informal and did not sound like human speech. On a more positive note, the actual content of the Facebook post was still relevant to the task. This trend was noticed across all queries [1].

Experiment 02: Translation

In this section, we particularly reflect on two samples of text translated from English to Sinhala: (1) a newspaper notice for a road closure, and (2) an opinion piece from a newspaper about addressing psychosocial risks in the workplace. English newspaper articles from a Sri Lankan newspaper were specifically selected to ensure that the text is set in the Sri Lankan context. This is to control for any errors that may creep in from mere context mismatch. The trends in the Sinhala output of this experiment closely followed those of the text generation task. The road closure notice, which took a formal tone and contained direct factual statements, was appropriately translated. The grammar, syntax, and tone were suitable for the text. The opinion piece, which contained complex syntax, metaphorical speech, and conversational tone, was translated into Sinhala in a stilted and formulaic tone. The limitations persisted as a trend across all queries and models used [2].

[1] Please see 'text-gen/text-gen-1.txt' in the GitHub repository for specifics on the Facebook post and 'text-gen/text-gen-2.txt' for specifics on the cover letter. All queries used for testing text generation are available in the 'text-gen' folder in the GitHub repository.

[2] Please see 'translation/translation-1.txt' in the GitHub repository for specifics on the opinion piece, and 'translation/translation-2.txt' for specifics on the road closure

Experiment 03: Information retrieval

For this experiment, we prompt the models to find information on topics that are relevant in the Sri Lankan context. All models were enabled to conduct a web search. We use two specific examples here for the discussion: (1) how to apply for a national identity card from a specific city, and (2) whether taxes must be paid to import a business sample to Sri Lanka. Across the models, responses were comprehensible, despite some errors in grammar and syntax. In the first query about applying for a national identity card, all model output contained hallucinated steps in the process. In the second query, the answer to the specific question was accurate (leaving aside all other contextual information that LLMs often provide), when compared with the information provided by the Sri Lanka Customs website as the source of truth. The information on the Sri Lanka Customs website is clearly provided in a Frequently Asked Questions page, under the question "Can I import a sample for my business without taxes?". In this task category, we notice a two-pronged challenge: (1) LLMs provide hallucinated information as facts; (2) the information space itself can be limited and poorly organised, preventing not only LLMs but even humans from finding the necessary information [3].

In the next section, we synthesise the findings of the above experiments to formulate the research question.

4 Problem statement and research question

Based on our experiments, we identify the following three key limitations in the practical usage of LLM-based conversational tools in Sinhala. Note that these are generalised from the output we have obtained from our experiments. Further discussion regarding this generalisation is provided in section 6.

Limitation 01: LLM conversational tools produce acceptable responses in formal Sinhala, but not in informal Sinhala.

Limitation 02: LLM conversational tools perform poorly when they do not have access to data from the information space relevant to Sinhala speakers.

Limitation 03: LLM conversational tools make errors in grammar, syntax, spelling, and writing.

Based on these limitations observed, improvements are needed in the linguistic capabilities of conversational systems and the information space that the systems can access.

As such, the research question we aim to answer is:

How can advancements in LLMs and RAG workflows be brought together to improve the practical value and usability of LLM-based conversational systems for Sinhala speakers?

In section 5, we discuss how we envision an agentic RAG system to address the research question and the limitations we identified.

5 Exploring agentic retrieval as a solution

In order to optimise LLM-based conversational systems for low-resource languages, we propose an agentic RAG framework that holistically brings together LLMs, RAG, and natural language processing.

The agentic RAG system contains three key components, as detailed below:

5.1 User interface

This is the interface where the user inputs their query in their chosen natural language. The interface will closely resemble an LLM-based chat interface, because this reduces the cognitive load on the user in using the system.

5.2 LLM agent

This is an LLM agent that receives the user query and decides the best possible mechanism to formulate a response. Depending on the query, the agent may respond with trained (parametric) knowledge, or may resort to utilising additional tools from the LRL-Toolkit outlined in section 5.3.

5.3 LRL-Toolkit

The LRL-Toolkit (Low-Resource Language Toolkit) is a set of natural language processing and LLM-based tools. At this stage of the study, we specifically envision three tools that address the three limitations that we identified through the experiments presented in section 3. The LRL-Toolkit includes a **retrieval tool** with access to external knowledge, addressing

notice. All queries used for testing translation are available in the 'translation' folder in the GitHub repository.

[3] Please see 'retrieval/retrieval-2.txt' in the GitHub repository for specifics on the identity card-related query and 'retrieval/retrieval-4.txt' for specifics on the taxes-related query. All queries used for testing information retrieval are available in the 'retrieval' folder in the GitHub repository.

the limitation of sparse parametric knowledge in LLMs for low-resource languages. We have considered a variety of underlying information architectures to support this tool, including semantic databases, knowledge graphs, and ontologies. We are particularly keen to investigate incorporating knowledge graphs and ontologies, as their applicability in low-resource contexts is discussed in detail in the literature (Kaffee et al., 2023; Vats et al., 2023; Lignos et al., 2022). We believe that building domain-specific repositories of knowledge (for example, government processes or legal or medical information) would provide a way to scale the system meaningfully. The **context tool** provides additional linguistic and cultural nuances to the LLM agent to decide how best to formulate the final output. For instance, diglossia is a feature of Sinhala, where a high variety of the language is used for formal and written Sinhala, while a low variety of the language is used in informal and conversational settings. As we noted in Limitation 01, LLMs tend to misrepresent the tone of speech when responding to diverse conversational queries. The purpose of the context tool is to ensure that such contextual information is available to the agent when drafting the response. The **orthography tool** supports the agent in drafting responses that align with the grammatical rules and conventions of writing in the language. We envision using rule-based natural language processing approaches, as we aim to incorporate more deterministic linguistic knowledge into probabilistic large language models.

Figure 1 is a diagrammatic representation of the conceptualised framework.

A key differentiator of this system, compared to conventional agentic systems used with higher resourced languages, is the context and orthography tools, which aim to ground the output in the cultural and linguistic nuances of the language.

5.4 System evaluation

We aim to evaluate the final implementation of the system with human evaluators, using question-answer pairs that can particularly test the three limitations we identified in section 4. Using human evaluators enables us to test for limitations related to diglossia, which is where conventional models have been observed to fail. We are cautious of using emerging approaches such as LLMs-as-a-judge, because evaluating subtle nuances in the tone of speech is not a task that we believe an LLM can reliably carry out in Sinhala.

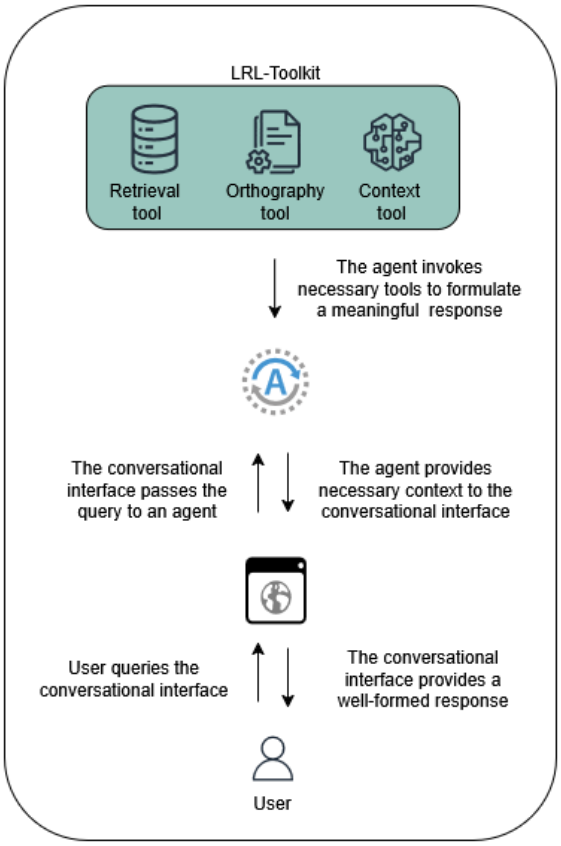

Figure 1: Diagrammatic representation of the proposed agentic RAG system

In our evaluations, we use the same question-answer pairs with a select sample of widely available LLM conversational tools and the implementation of the proposed system, where performance on widely available tools forms the baseline. The evaluation rubrics include factual accuracy, comprehensiveness, and precision to measure the quality of information retrieval, and grammar, syntax, lexical choice, and orthography to measure linguistic performance.

6 Limitations and mitigation strategies

In this section, we discuss two limitations of our work, and lay out the planned way forward.

In the experimental setup of section 3, we have used a select set of examples from three widely available LLM-based conversational tools to derive our findings. These limitations broadly align with challenges noted in the literature and the practical and anecdotal evidence provided by Sinhala speakers. However, we have not conducted a sys-

tematic evaluation with multiple annotators. While we believe our findings at this stage are sufficient to conceptualise the limitations and the retrieval framework, as the study progresses, we aim to do wider benchmarking to further refine the problem.

As this study and the system are in initial exploratory stages, the proposed system conceptualisation is subject to change to adapt to the evolving needs of the work, as well as to integrate the latest developments in LLMs. As a result, we may need to modify the proposed components as we develop and evaluate our work. Presenting this exploratory work for early feedback helps us shape our future work, and makes us aware of any potential drawbacks that we may have overlooked in our conceptualisation of system design.

References

Mitha Alshammary, Md Nahiyan Uddin, and Latifur Khan. 2025. Arcadia: A scalable arabic rag framework with hierarchical and context-aware retrieval. In *2025 IEEE 8th Congress on Information Science and Technology (CiSt)*, pages 238–243.

H.W.K. Aravinda, Rashad Sirajudeen, Samith Karunathilake, Nisansa de Silva, Rishemjit Kaur, and Surangika Ranathunga. 2025. Sinllama - a large language model for sinhala. In *2025 Moratuwa Engineering Research Conference (MERCon)*, page 617–622. IEEE.

William Christian, Daniel Adamlu, Adrian Yu, and Derwin Suhartono. 2025. Bridging language gaps with adaptive rag: Improving indonesian language question answering.

Shreyas Joshi, Muhammad Shahnawaz Khan, Aditya Dafe, Kavita Singh, Vedant Zope, and Tanish Jhamtani. 2024. Fine tuning llms for low resource languages. In *2024 5th International Conference on Image Processing and Capsule Networks (ICIPCN)*, pages 511–519.

Lucie-Aimée Kaffee, Russa Biswas, C. Maria Keet, Edlira Kalemi Vakaj, and Gerard de Melo. 2023. Multilingual Knowledge Graphs and Low-Resource Languages: A Review. *Transactions on Graph Data and Knowledge*, 1(1):10:1–10:19.

Nir Kshetri. 2024. Linguistic challenges in generative artificial intelligence: Implications for low-resource languages in the developing world. *Journal of Global Information Technology Management*, 27(2):95–99.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen tau Yih, Tim Rocktaschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *ArXiv*, abs/2005.11401.

Constantine Lignos, Nolan Holley, Chester Palen-Michel, and Jonne Sälevä. 2022. Toward more meaningful resources for lower-resourced languages. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 523–532, Dublin, Ireland. Association for Computational Linguistics.

Edmund V. Ndimbo, Qin Luo, Gimo C. Fernando, Xu Yang, and Bang Wang. 2025. Leveraging retrieval-augmented generation for swahili language conversation systems. *Applied Sciences*, 15(2).

Hellina Hailu Nigatu, Atnafu Lambebo Tonja, Benjamin Rosman, Thamar Solorio, and Monojit Choudhury. 2024. The Zeno's paradox of 'low-resource' languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17753–17774, Miami, Florida, USA. Association for Computational Linguistics.

Andrew Smart, Ben Hutchinson, Lameck Mbangula Amugongo, Suzanne Dikker, Alex Zito, Amber Ebinama, Zara Wudiri, Ding Wang, Erin van Liemt, João Sedoc, Seyi Olojo, Stanley Uwakwe, Edem Wornyo, Sonja Schmer-Galunder, and Jamila Smith-Loud. 2024. Socially responsible data for large multilingual language models.

Atnafu Lambebo Tonja, Bonaventure F. P. Dossou, Jessica Ojo, Jenalea Rajab, Fadel Thior, Eric Peter Wairagala, Anuoluwapo Aremu, Pelonomi Moiloa, Jade Abbott, Vukosi Marivate, and Benjamin Rosman. 2024. Inkubalm: A small language model for low-resource african languages.

Preeti Vats, Nonita Sharma, and Deepak Kumar Sharma. 2023. Hkg: A novel approach for low resource indic languages to automatic knowledge graph construction. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* Just Accepted.

Tianyang Zhong, Zhenyuan Yang, Zheng Liu, Ruidong Zhang, Yiheng Liu, Haiyang Sun, Yi Pan, Yiwei Li, Yifan Zhou, Hanqi Jiang, Junhao Chen, and Tianming Liu. 2024. Opportunities and challenges of large language models for low-resource languages in humanities research. *ArXiv*, abs/2412.04497.