Accessible Tamil Sangam Literature using Artificial Intelligence

# Introduction

Tamil is one of the longest surviving classical languages spoken by Tamil people in South Asia and Tamil diaspora across the world.

Tamil Sangam literature (also known as the poetry of the noble ones) is the earliest known literature of South India. The Sangam literature era spanned from c.300 BCE to 300CE. Tamil Sangam literature is broadly classified into Akam (அகம், inner) and Puram (புறம், outer). Agam literature focuses on feelings/emotions of romantic love, ethics, morality, nature and similar ones. Puram literature focuses on heroic deeds of war and public life. These literatures also serve as historical documents for many notable events and people, with vivid description.

One of the literatures that is most significant of all is Thirukural (திருக்குறள்). Thirukural contains 1330 short couplets (4 words in the first line and 3 words in the second line) which focuses on moral and ethics, and it is one of most translated ancient literature works.

In short, Tamil literature is a treasure trove of poems that reflects vibrant Tamil culture, inclusivity of both men and women in Tamil society, ethical and moral values upheld by them and anecdotes of historical events and people.

Anyone who understand these literatures can learn many virtues and improve their life.

However, Tamil we use today has changed significantly over time and it is not possible to understand/enjoy Tamil Sangam literature by anyone but Tamil scholars. This is mainly due to the changes to vocabulary, most words used in Tamil Sangam literature is not used in modern Tamil and things like metaphors, similes, and imageries used in the Tamil Sangam literature for poetic elegance requires historical and contextual knowledge to understand and uncover the true meanings.

We have LLMs that can take prompts in modern Tamil and respond in modern Tamil, upon evaluation, they are significantly inadequate when using them to understand/translate Tamil Sangam literature in modern Tamil.

The aim of this study is to explore the possibility of making Tamil Sangam literature accessible to anyone who can read and understand modern Tamil using cost-effective fine-tuning techniques and evaluate their performances.

Currently, the interest in Tamil literature work is fading, mainly because English dominates the internet and almost all technological advancements. This research would bring latest technological advancements to Tamil too, thereby, reviving interest in Tamil literature by making it accessible to modern Tamil people.

The outcome of this study is fine-tuned open-source Tamil LLMs that takes Tamil Sangam literature work as prompt and output the explanation in modern Tamil at inference time.

The dataset curated for this work is made available for anyone to use for their own research.

The quality of the outcome is verified by combining two different approaches - automatic evaluation using ROUGE (Lin, 2004) and BLEU (Papineni et al., 2002) metrics [TODO: ADD OTHER METHODS HERE] and Human verification to ensure quality, coherence and simplicity.

The fine-tuned models’ performance and their comparisons and limitations are included.

# Related Work

This research work is built on top of Tamil LLaMA, an excellent work by Balachandran, 2023, and Tamil Mistral by Hemanth Kumar, 2024.

Tamil LLaMA was created by enriching LLaMA with additional 16,000 Tamil tokens and then by fine-tuning using Tamil instructions which are translated subset of Alpaca (Taori et al., 2023) dataset and OpenOrca (Lian et al., 2023) dataset.

Tamil Mistral (Hemanth Kumar, 2025) was created by fine-tuning Mistral-7B-Instruct (Jiang et al., 2023) modal using 400k Tamil instruction dataset.

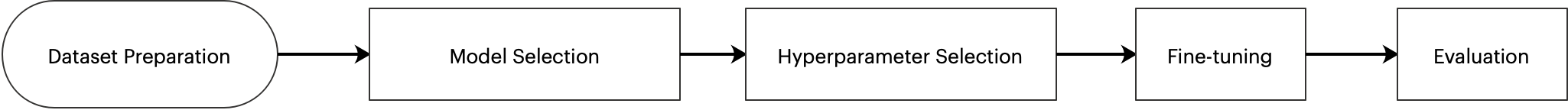
Although, Tamil LLaMA (Balachandran, 2023) and Tamil Mistral (Hemanth Kumar, 2025) are good at taking Tamil prompts and returning Tamil completion at inference time, they do not have sufficient understanding of the Tamil language to explain Tamil Sangam literatures.

[TODO: add other low resource language work]

However, we now know from the work by both Balachandran, 2023 and Hemanth Kumar, 2025 that pre-trained LLMs can successfully be fine-tuned for Tamil language. This gives us a strong foundation and high confidence that it is worth exploring fine-tuning these models on Tamil Sangam literature instruction dataset.

# Methodology

[TODO Intro]



## 3.1 Models

Open source LLMs make research and product development in any field/area that you can imagine possible without incurring huge cost and time, usually involved in pre-training. However, the availability of models of various size and capability makes choosing a pre-trained model, for the problem at hand, a significant task. Choosing a right model could result in significant increase in performance and quality while reducing the time and cost involved in prompt engineering and fine-tuning.

To evaluate the feasibility of this task (explaining Tamil Sangam Literature in modern Tamil), we need a model that is ready to be experimented with. Because of this and the requirement of a model to work with Tamil tokens, the number of LLMs available to select for this task have been reduced significantly.

Tamil LLaMA Instruct 7B (Balachandran, 2023) and Tamil Mistral Instruct 7B (Hemanth Kumar, 2025) were chosen as models for this study because they satisfy both language and ready-to-use requirements.

## 3.2 Dataset

For the dataset, it was not feasible to collect and include entire Tamil Sangam Literature corpus so, again due to time constraints, we only picked Thirukural, Agananuru and Purananuru corpus for instruction fine-tuning.

Thrikural contains 1330 couplets and each having explanation written in modern Tamil by 3 different well respected Tamil scholars. We have created this dataset from Thirukural Karpom website and used all of the available explanations for each of the couplets.

Agananuru contains 400 poems, and Purananuru contains 398 poems (2 are missing). The poems are of varying size. These poems, their explanation, and contextual data are taken from Tamil Drops website.

[TODO: explain how augmented data created using GPT – just ask and few shot prompting]

[TODO: explain manual verification approach and effort]

We then kept 10% of the data as holdout dataset for testing and used the remaining 90% training. Though this is a very small corpus for training, it is a good starting place.

## 3.3 Technique and Hyper Parameters

We have instruction fine-tuned both Tamil LLaMA and Tamil Mistral (7b) models with the same dataset. Fine-tuning was done using one of the PEFT methods - QLoRA (Dettmers et al., 2024) method with rank=64, alpha=16 and dropout=10%. NVIDIA A100, 40GB GPU was used.

Both Tamil LLaMA and Tamil Mistral are fined-tuned with ground truth and augmented datasets, resulting 4 different models to compare.

### 3.3.1 Hyper Parameters

The following hyper parameters are used for fine-tuning both Tamil LLaMA 7B and Tamil Mistral 7B

Optimiser = Adam 2

Learning rate = 2e-4

Weight Decay = 0.001

Training Epoch = 10

[TODO: explain quantisation used in detail]

We did not experiment with different QLoRA parameter options due to time and compute constraints. Training was done for 10 epochs but we noticed that the training loss was not going down after 5 epochs in both Tamil LLaMA 7B and Tamil Mistral models, this is probably due to small training data size.

### 3.3.2. Prompt template

[TODO: explain the prompt template used]

## 3.4 Performance

We have run the inference pipeline for the validation dataset and the inference time was very good when using A100 GPUs (under 3 seconds per instruction). Optimising inference time is one of the tasks That need exploring further.

Fine-tuned models can be found at

1. <https://huggingface.co/kodebot/tamil-llama-7b-instruct-tamil-poem-fine-tune-web-only-data-attempt-1-v0.1>
2. <https://huggingface.co/kodebot/tamil-llama-7b-instruct-tamil-poem-fine-tune-web-aug-attempt-1-v0.1>
3. <https://huggingface.co/kodebot/tamil-mistral-7b-instruct-tamil-poem-fine-tune-web-only-data-attempt-1-v0.1>
4. <https://huggingface.co/kodebot/tamil-mistral-7b-instruct-tamil-poem-fine-tune-aug-data-attempt-1-v0.1>

Dataset prepared for this work can be found at

1. <https://huggingface.co/datasets/kodebot/Thirukural_tamil_with_meaning>
2. <https://huggingface.co/datasets/kodebot/Thirukural_tamil_with_meaning_aug>
3. <https://huggingface.co/datasets/kodebot/Agananuru_Tamil_with_meaning>
4. <https://huggingface.co/datasets/kodebot/Agananuru_tamil_with_meaning_aug>
5. <https://huggingface.co/datasets/kodebot/Purananuru_Tamil_with_meaning>
6. <https://huggingface.co/datasets/kodebot/Purananuru_tamil_with_meaning_aug>

LLaMA 2 finetuning pipeline for fine-tuning was developed loosely based on great work from Labonne, 2023.

# Evaluation

[TODO]

The quality of inference text was evaluated using ROUGE (Lin, 2004) score. The ROUGE evaluation score from test dataset was not great, we got 0 for all variations (rouge1, rouge2, rougeL and rougeLsum) of the score.

[TODO: bertscore]

[TODO: chrf]

[TODO: meteor]

[TODO: sacrebleu]

[TODO: manual verification]

|  |  |
| --- | --- |
| A graph of different colored bars  AI-generated content may be incorrect. | A graph of different colored bars  AI-generated content may be incorrect. |
| A graph of different colored bars  AI-generated content may be incorrect. | A graph of different colored bars  AI-generated content may be incorrect. |

However, when verified manually, the Tamil text completion (generated text) from instruct model is somewhat related to the input text, although they are not coherent, grammatically correct, contains some unexpected non-Tamil tokens and many repeated tokens or phrases, it is the evidence that this approach is in the right direction.

Although this is not a great result, but based on the understanding of how LLMs work, this is something that we can improve by obtaining larger corpus of data for fine-tuning. The bigger the corpus, the better the result would be.

# Limitations and Conclusions

[TODO: Review]

Also, no changes were made to tokeniser as part of this study due to time constraints but this needs to be explored further.

This study was also very limited in terms of the choice of model’s architecture. LLaMA uses decoder only architecture (Touvron et al., 2023) also known as Autoregressive models. Autoregressive models are good at text generation and a good choice of summarisation (Touvron et al., 2023), however, if we consider explainer of ancient Tamil poems in modern Tamil as a kind of translation, then decoder only models may not be a great choice and we could do better with the use of encoder. So, for translation tasks, full transformer models like BART (Lewis et al., 2019) or T5 (Raffel et al., 2020) that implements ground-breaking innovation presented in “Attention is all you need” paper (Vaswani et al., 2017) could be worth exploring.

Again, due to time constraints, in-context learning and other prompt engineering techniques were not explored for this pilot study but this is something that is worth exploring in the implementation stage.

# References

Askell, A., Bai, Y., Chen, A., Drain, D., Ganguli, D., Henighan, T., Jones, A., Joseph, N., Mann, B. and DasSarma, N. 2021. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*.

Balachandran, A. 2023. Tamil-Llama: A New Tamil Language Model Based on Llama 2. *arXiv preprint arXiv:2311.05845*.

Cui, Y., Yang, Z. and Yao, X. 2023. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*.

Dettmers, T., Pagnoni, A., Holtzman, A. and Zettlemoyer, L. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*. **36**.

Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L. and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Kudo, T. and Richardson, J. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V. and Zettlemoyer, L. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Lian, W., Goodson, B., Pentland, E., Cook, A., Vong, C. and Teknium 2023. OpenOrca: An Open Dataset of GPT Augmented FLAN Reasoning Traces. *HuggingFace*. [Online]. [Accessed 23 February 2024]. Available from: https://huggingface.co/datasets/Open-Orca/OpenOrca.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries *In*: *Text summarization branches out*., pp.74–81.

Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation *In*: *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*., pp.311–318.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W. and Liu, P.J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*. **21**(1), pp.5485–5551.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A. and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P. and Hashimoto, T.B. 2023. Stanford alpaca: an instruction-following llama model (2023). *URL https://github. com/tatsu-lab/stanford\_alpaca*.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P. and Bhosale, S. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*. **30**.