Tools for constructing AI/ML solutions in Tamil

Authors: Abdul Majed Raja RS <1littlecoder@gmail.com>, Muthiah Annamalai† <ezhillang@gmail.com>

Abstract

We contend creation of new AI/ML applications in Tamil is still hard despite relative abundance of Tamil datasets [1]; this is due to scarcity of Tamil tools. However the accessibility of fully-trained models and capability of providing pre-trained models, like huggingface [2], are much harder and still require domain expertise in hardware and software.

While individuals have have published [3-4] some small Jupyter notebooks, and articles, but they still remain inadequate to scale the breadth of Tamil computing needs in AI world among:

(1) NLP – Text Classification, Recommendation, (2) Spell Checking, (3) Correction tasks, TTS – speech synthesis tasks, and ASR – speech recognition

While sufficient data exist for 1, the private corpora for speech tasks (அருந்தமிழ் பட்டியல்), the public corpora of a 300hr voice dataset recently published from Mozilla Common Voice (University of Toronto, Scarborough, Canada leading Tamil effort [5a]) have enabled data completion to a large degree for tasks 2 and 3. Private repositories exist for voice data under Penn LDC.

Ultimately the missing tooling can provide capability to quickly compose AI services based on open-source tools and existing compute environment to host services and devices in Tamil space. We propose for community to build a pytorch-lightning [5b] like API for Tamil tasks across NLP, TTS, ASR via AI so that newer AI/ML applications are easily built. Role of central institutions and governments is also explored.

1 Introduction

Recently, DALL-E images (Generative AI) by Open-AI, and Stable Diffusion models by Emad Mostaque of Stability AI provides promise generative capabilities to average users unleashing creativity (Fig. 1). These tools and technologies provide pathways to adapt some fast AI applications for good (provide TTS in voice of disabled person who has lost voice) and nuisance, or mischief (fake-news) etc. Generative AIs have their unresolved problems we list under biases portion of this paper.

AI technologies allows several applications for Tamil community but we have report that adapting them safely and creatively with positive outcomes require more work from side of tooling, data curation among other metrics.

Fig. 1: Prompt to DALL-E from OpenAI [6] describing (a) street temple-car festival (Thiruvizha) in Madurai at night; (b) Tamil family celebrating festival of lights Deepavali in Madurai; same for bottom row as well. The prompt asked AI to generate in pastel style.

2 Models

Traditionally, Machine Learning Models were built for specifics - Specific Task, Specific Language - like Text Classification for English, Text Classification for Tamil and so on. Recently, Since the rise of Transformer-based models like BERT, The lines of these specifics have gotten blurred. Thanks to Large Language Models that are trained on huge datasets and Millions and Billions of Parameters, The same model that's used for English Translation can also be used for Tamil Translation.

2.1 Zero-Shot Usage

Zero-shot learning is a machine learning technique that allows a model to recognize an object or a concept that it has never seen before. While many, if not most, machine learning models require a large amount of training data, zero-shot learning can recognize an object or concept without any new training data. Large Language Models are trained on a large amount of text data. These models can be used to answer questions about text, such as "what is the most

likely next word in a sentence?" Hence these models work fairly good out-of-box making them ideal candidates for a good Zero-shot usage.

An example of a Zero-shot usage is to use a Large Language Model like GPT-3 for Sentiment Classification for Tamil Language. Thus eliminating the need for new training data.

2.2 Model Fine-Tuning

Model Fine-tuning is a technique that allows a model to be trained on a new dataset. Large Language Models or Foundational Models can be fine-tuned to get improved performance on a new dataset. This became quite popular since the beginning of Transfer Learning. Transfer Learning is the process of applying knowledge gained in one context to a different context. For example, if you have learned how to use Microsoft Word, you can apply that knowledge to using OpenOffice. In the same way, Foundational Models or Large Language Models trained for Text Generation tasks can be used for other applications like Sentiment Analysis, Entity Extraction, Grammar Correction and so on.

While Zero-shot Learning can work fairly well in a general context and is good for the English language, It can improve the performance of these models very well if they get trained on a relatively smaller dataset. Fine-tuning a Large Language Model to let the fine-tuned model perform NLP for the dataset that is similar to the fine-tuned dataset can be a very effective way to use Foundational LLMs for Tamil NLP tasks.

This does not mean that these models cannot be used in Zero-shot capability but means they do a lot better if they are fine-tuned on relevant dataset which in our case is new Tamil Dataset.

2.3 Model Serving

One of the least addressed problems in ML and AI is how to serve the Model to developers and end-users. It is important that we serve both Developers, who would build on top of our toolkit and end-users who would directly use our toolkit to leverage AI/ML for their Tamil NLP requirements. Hence we propose two distinct ways to serve these models as a central toolkit for Tamil AI/ML

1. A Python Library for Developers
2. A Gradio App

The Python Library that can be hosted for free on PyPi can serve the Tamil developers who want to use our Toolkit to build applications and services leveraging Tamil AI/ML while the Gradio App that can be run locally on any computer (preferably GPU) or hosted for free on Hugging Face Spaces can serve the end-users like Tamil Content Creators who want to include our Toolkit as part of their workflow.

2.4 Model Selection

While there is a growing number of Large Language Models every single day, It's very important for us to pick the right model that can work well for Tamil Language. One of the easiest ways to select the right model for Tamil is by looking at the training dataset information.

Most open source Large Language Models indicate their training dataset composition. From that information, We can understand which of those existing Large Language Models have got the most Tamil Data during the Model Training. This is primarily applicable for a Zero-shot Learning since Fine-tuned models mostly would have been fine-tuned on Tamil Dataset.

For example, Big Science's BLOOM model was 46 Natural Languages and 13 Programming Languages. Tamil is one of those Natural Languages of the Indic category which is ~4% of all the languages. Even though Tamil is a very small part of the entire language set, The Zero-shot tasks like Text Generation that we experiment for Tamil works fairly fine.

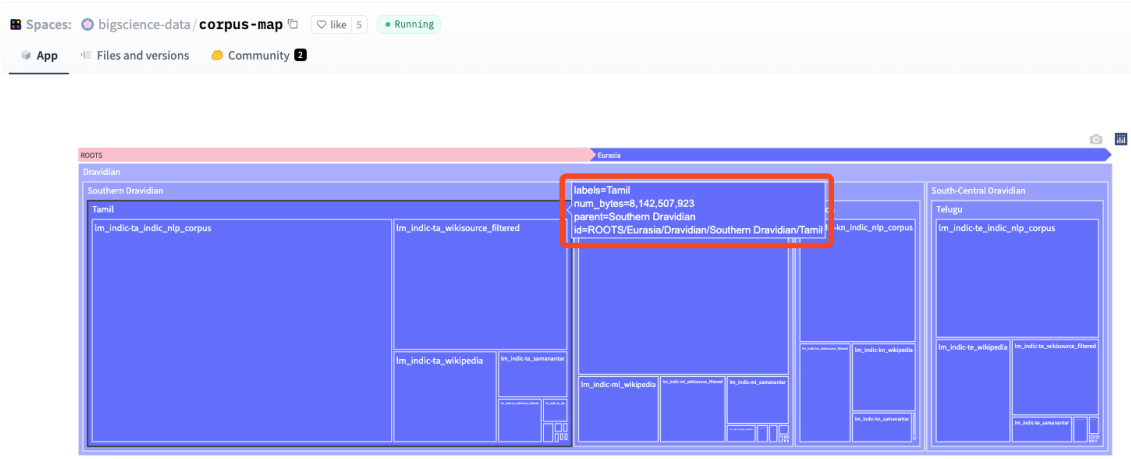

Fig. 2: Corpus map used to train a specific model [7]

BLOOMZ and mT0, a family of models capable of following human instructions in dozens of languages zero-shot. BLOOMZ and mT0 are finetuned from BLOOM and mT5 pretrained multilingual language models on crosslingual task mixture (xP3) and the resulting models are capable of crosslingual generalization to unseen tasks & languages. In the case of BLOOMZ & mT0, Tamil is just 0.5% of the fine-tuned data, Yet the model is capable of performing tasks like Sentiment analysis, Text generation, Keywords creation and so on.

3 AI Applications for Tamil

RoBERTa and BERT models are customized for Tamil by finetuning the final layers for classification of idioms in work [17]. We report in this section how various NLP, TTS applications can be solved using AI/ML models.

3.1 Spelling Correction with LLM

We may use masked words as '<mask>' when input sentence to check for spelling correction on certain words in sentence that are out-of-dictionary or not correctable by known rules [8];

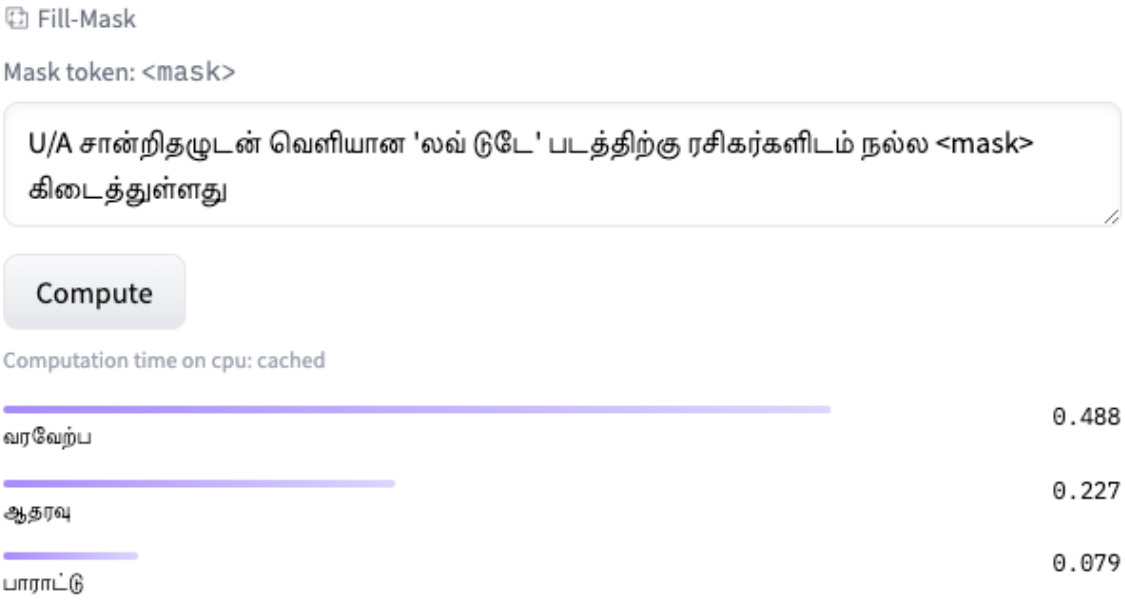

Fig. 3.1: Spelling checker functionality of LLM using masking; missing word is recommended வரவேற்பு.

3.2 Sentiment Recognition with LLM

Sentiment Recognition in NLP is the task of identifying the correct sense of a word in a given context. This is one of the most used tasks in NLP given how much text data is available in the world. It's also largely sought after given the business applications of Sentiment Analysis Models.

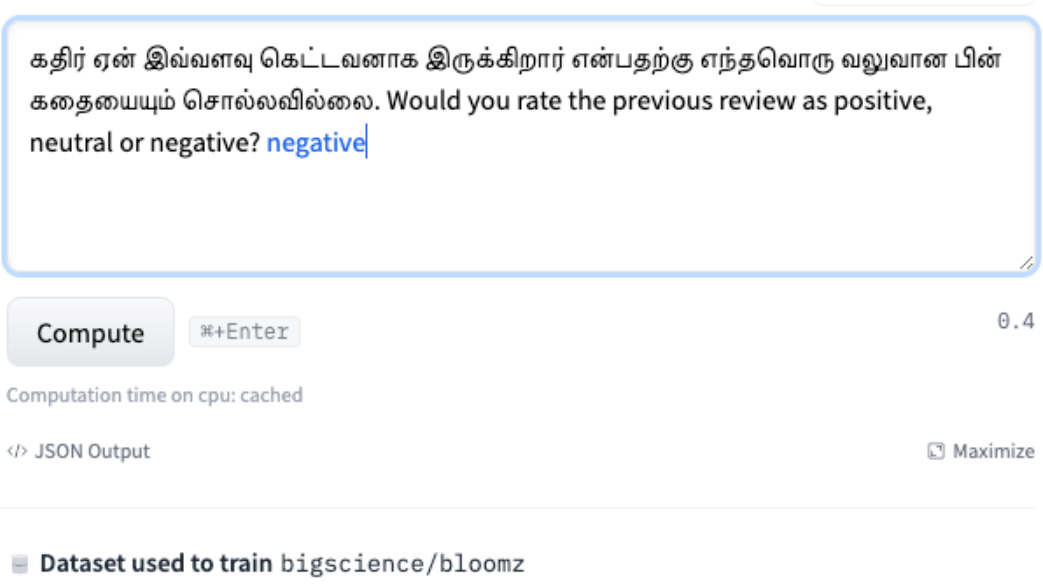

Fig. 3.2: sentiment recognition of text by using LLMs.

With the help of LLMs, We can use the existing Foundational models for Sentiment Analysis in Tamil Language without the need for a new training dataset. For example, We used **BLOOMZ** LLM for performing Sentiment Analysis of a Tamil Review in a Zero-shot Context.

3.3 Named-Entity Recognition with LLM

Named-Entity Recognition is the task of identifying the names of people, places, organizations, and other entities of interest in text. This is a key component of many natural language processing applications. Using Large Language Models for Named-Entity Recognitions can be a very good application.

Below is an example of using BLOOMZ model for Named-Entity Recognition.

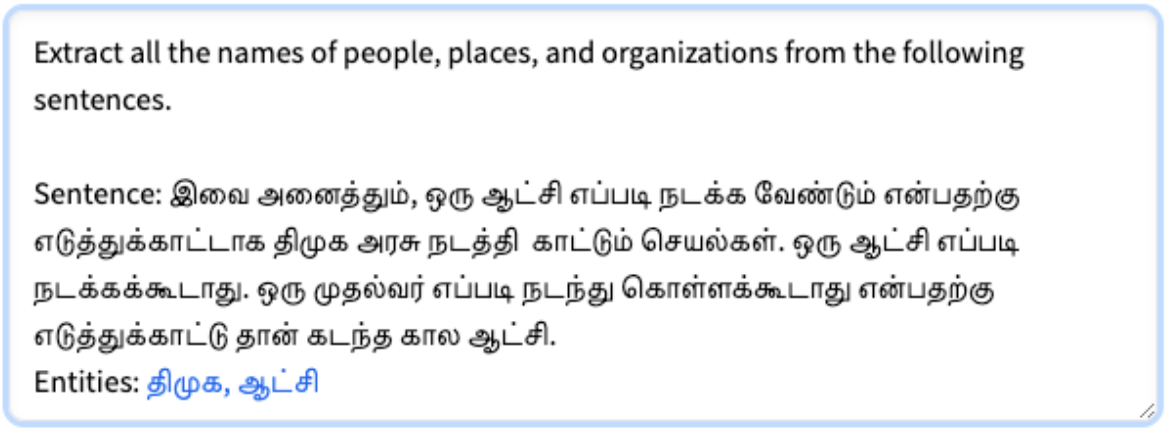

Fig. 3.3: Name-entity recognition using LLMs.

3.4 Audio and Voice Applications - ASR, TTS

ASR and TTS models based on sequence-to-sequence transformation pioneered by researchers at Meta (Facebook) have been adopted by authors to present a good demonstrations of TTS applications in Tamil, and other major Indian languages [15]. We note however number to words conversion remains a sore point in this implementation as compared to work [20].

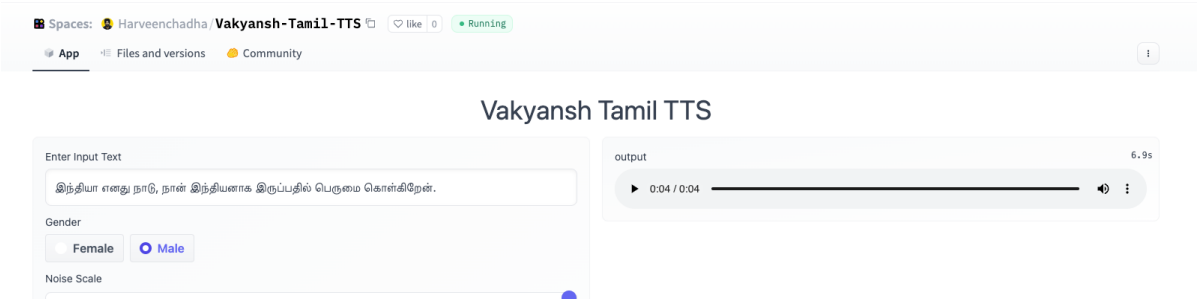

Fig 3.4: Demo Space for work [15]
https://huggingface.co/spaces/Harveenchadha/Vakyansh-Tamil-TTS

Clearly we can see the improved quality of AI/ML based TTS over unit-selection synthesis based approaches.

OpenAI's Whisper [16], as reported in [18], is demonstrated to translate high-quality lyrical Tamil audio with transcription and errors highlighted in the following figure.

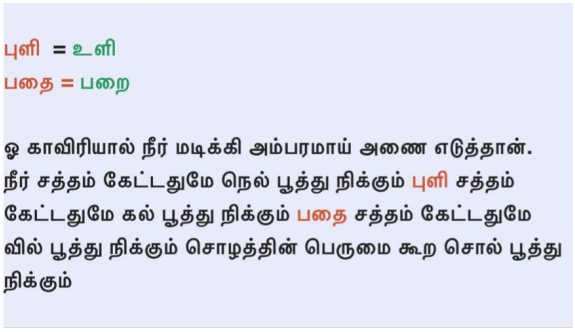

Fig 3.5: Experimental results of Malaikannan [18] using OpenAI Whisper for Tamil ASR based on the popular song "பொன்னி நதி" from the movie "பொன்னியின் செல்வன்." This is showing low word-error rate 6.4%.

4 Tamil Tooling gaps

Our proposal is the following to address the gaps, and we also understand many of the steps are further problems on their own:

1. Develop a open-source toolbox for pre-training and task training specialization
2. Identify good components to base effort
3. Contribute engineering effort, testing, and validation
    1. R&D – DataScience, Infra, AI framework
    2. Engineering Validation – DataScience, Tamil language expertise
    3. Engineering – packaging, documentation, distribution
    4. Project management
4. Library to be liberally licensed MIT/BSD
5. Open-Source license for developed models
6. Find hardware resources for AI model pre-training etc.
7. Managed by a steering committee / nominated BDFL
8. Scope – decade time frame
9. Financial support for such a wide effort

4.1 Datasets related Tooling

Currently hosted datasets [1] not consumable in uniform interface for Torch or with TensorFlow in a uniform format; we have only raw data today.

4.2 Model Related Tooling

- model attribute, training time, standardized accuracy metrics, training dataset, notions of biases etc. are absent

4.3 Compute related challenges

Free compute is limiting on what can be done; Google Cloud CoLaboratory is limited in credits that are freely available; training CNN or LSTM takes lot of time on laptop scale hardware.

There is a chronic need for special purpose AI Accelerators (GPU, RDUs, etc.) for large scale models pre-training; there needs to be efforts in private-public collaboration to subsidize cost and sponsorship these activities.

4.4 Problems and Biases

Just a decade ago the auto-complete in Google search query with the words “Tamil “ will always end with “Tigers,” limiting what an uninformed lay-person could learn about Tamil people, language or culture; which such a subjective bias has been removed it remains largely un-tested in various areas. This would be considered as harmful bias against Tamils by virtue of language marker in the discourse of [10].

Large language models (LLMs) are known to have problems with representing minorities along various margins, problems with performing math (calculators), potential to be environmentally harmful, repeat harmful stereotypes on minorities by age, nationality, race or other marking criteria [10], etc.

Language models exhibit a variety of expertise to work as auto-pilots in coding tasks [11], as email marketing assistants [12] etc. however as autonomous agents still much remains to be achieved [13] - current generation of AI models and agents are in rung-1 of 3-step ladder of causation [14] and act based on observation but not in a causal framework of learning which would be the creation of near-human level intelligence.

Specifically for Tamil language, as a largely under-resourced language, we find the nature of AI-systems to largely dependent on public data sets (uncurated) and few private data sets, and goodwill of giant corporations like Goolge or Meta (Facebook) to develop models for tasks. In such cases the pre-trained models are not qualified for biases. Additionally where data is not available or incorrect data is available the systems will not be able to reason correctly causing problematic consequences for applications of such AI models for Tamil community. Overall sufficient availability of compute, data, correctness and bias measures for Tamil tasks are needed to quantify bias in AI models.

Advent of generative AI models like DALL-E, Stable Diffusion etc. have created a chaotic situation of attribution, fair-use and copyright.

As a Tamil community we would want our real-world language, cultural, audio-visual, written and oral cultural milieu to be within the “in-distribution” of training set of the language/visual/multi-modal models for AI. When such a ecosystem of data driven AI modeling, and harm reducing systems exist perhaps someday we can hope to eliminate biases about individuals, groups, or minorities (by various labels) for creation of a oracular AI agents which can be native to Tamil.

5. Summary and Conclusions

AI/ML systems rely of good data; we note dominance of Tamil data reflects in metrics like OpenAI's Whisper (ASR model) performing on Tamil audio to have lowest word-error rate (at 20.6%) among Indian languages (even compared to Hindi at 26.9%) perhaps evidence of data prevalence and seeds of digitization and open-content in parallel corpora (audio + transcribed text) available in Tamil [16].

We have presented various aspects of AI/ML systems which can benefit the Tamil community in general and gaps in tooling which can accelerate the delivery of AI based applications in hands of general developer and community members, democratizing AI.

References:

1. INFITT அருந்தமிழ் - Awesome Tamil resource list, https://github.com/INFITTOfficial/awesome-tamil (accessed Nov , 2022).
2. T. Wolf, "Huggingface's transformers: State-of-the-art natural language processing," (2019).
3. M. Annamalai, "AI and Tamil Computing opportunities", tutorial at Tamil Internet Conference (2021) link
4. (a) AbdulMajedRaja Bloomz model for AI, https://www.youtube.com/1littlecoder (accessed Nov 14, 2022);
   (b) Niklas Muennighoff, et-al, "Crosslingual Generalization through Multitask Finetuning," arXiv:2211.01786 (2022)
5. (a) UTSC Tamil Digital Studies Program Common Voice project https://tamil.digital.utsc.utoronto.ca/en/tamil-common-voice
   (b) Pytorch Lightning https://www.pytorchlightning.ai/
6. DALL-E - Generative AI images from text by Open-AI, 2022 (accessed Nov 1, 2022)
7. Tamil portion of Corpus map of BigScience model, https://huggingface.co/spaces/bigscience-data/corpus-map (accessed Nov 28, 2022)
8. M. Annamalai, T. Shrinivasan, "Algorithms for certain classess of Tamil spelling correction," Tamil Internet Conference, Chennai, India (2019).
9. R. Bommasani et-al, "On the Opportunities and Risks of Foundation Models," Stanford Center for Research on Foundation Models Report, August (2021).
10. Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜" Proc. of ACM Conference FAccT '21, New York, NY, USA, pages 610–623, (2021)
11. Github Autopilot, https://github.com/features/copilot (accessed 2022)
12. Jasper AI (https://www.jasper.ai) , (2022)
13. Pearl, Judea, and Dana Mackenzie. "AI can't reason why." Wall Street Journal (2018).
14. Pearl, Judea, and Dana Mackenzie, "The Book of Why: The New Science of Cause and Effect," Basic Books, (2018).
15. Harveen Singh Chadha, et-al, "Vakyansh: ASR Toolkit for Low Resource Indic languages," arXiv:2203.16512 [cs.CL] (2022).
16. Alec Radford, et-al "Robust Speech Recognition via Large-Scale Weak Supervision," OpenAI Report (2022).

17. Briskilal, J. and Subalalitha, C.N., 2022. An ensemble model for classifying idioms and literal texts using BERT and RoBERTa. Information Processing & Management, 59(1), p.102756.
18. Malaikannan S, private communication (Nov, 2022).
19. (a) Malaikannan S, “Can a machine write a story ?,” blog post (2016).
(b) Anderj Karpathy, “The Unreasonable Effectiveness of Recurrent Neural Networks,” (2015).
20. Annamalai, Muthiah, and Sathia Mahadevan. "Generation and Parsing of Number to Words in Tamil." Tamil Internet Conference (2020).