Building Tamil AI

Open challenges and the role of the community

Abinaya Mahendiran

Oct 13, 2023

Abinaya Mahendiran

- CTO, Nunnari Labs,
- Program Manager, IITM
- M.Tech IT, International Institute of Information and Technology Bangalore
- Volunteer at AI Tamil Nadu, WTM, Data Conversations, GHCI, WAI, Women Who Code
- Interests: Building NLP/NLU/NLG/MLOps/Gen AI systems, Open source, Applied Research

https://abinayam02.github.io

@freakynut

https://medium.com/@abinayamahendiran

https://www.linkedin.com/in/abinayamahendiran/

https://topmate.io/abinaya_mahendiran

Agenda

Why Tamil AI?

Natural Language Processing

NLP: Techniques

NLP: Data

NLP: Pipeline

AI Tools using NLP

NLP for Indic Langauges

Open Research Challenges

Data Curation

Types of AI

Traditional Approach

Transfer Learning

Foundation Models

LLM Training

Prompt Engineering

Limitations of Gen AI

Gen AI: Research Directions

NLP Timeline

Generative AI for Tamil

Role of the community

Why Tamil AI?

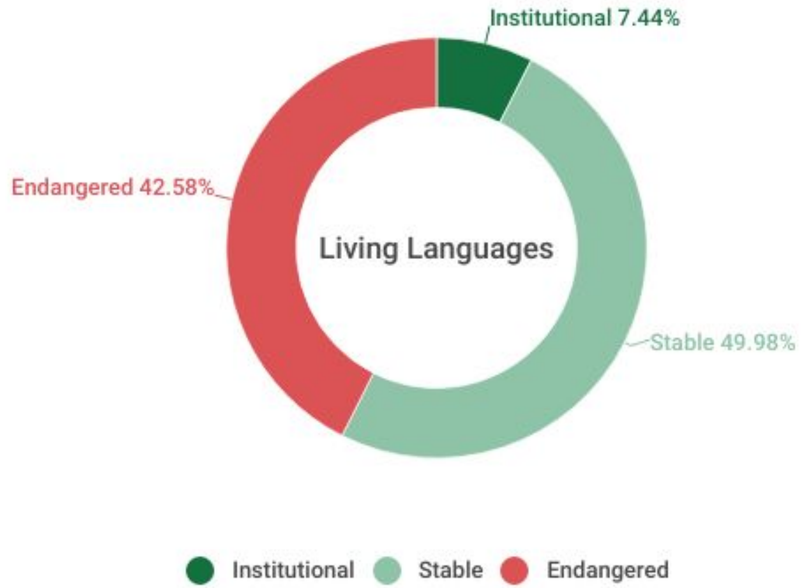
- Tamil is one of the oldest and longest surviving classical languages of the world - 80 million people speak Tamil

- Building NLP tools with the abundant literature available in Tamil can help our language withstand the technological evolutions.

- Tamil AI tools can help in spreading the cultural and literary aspects of our language to the next generation.

- Building a knowledge repository of our language is essential to represent the global southern languages.

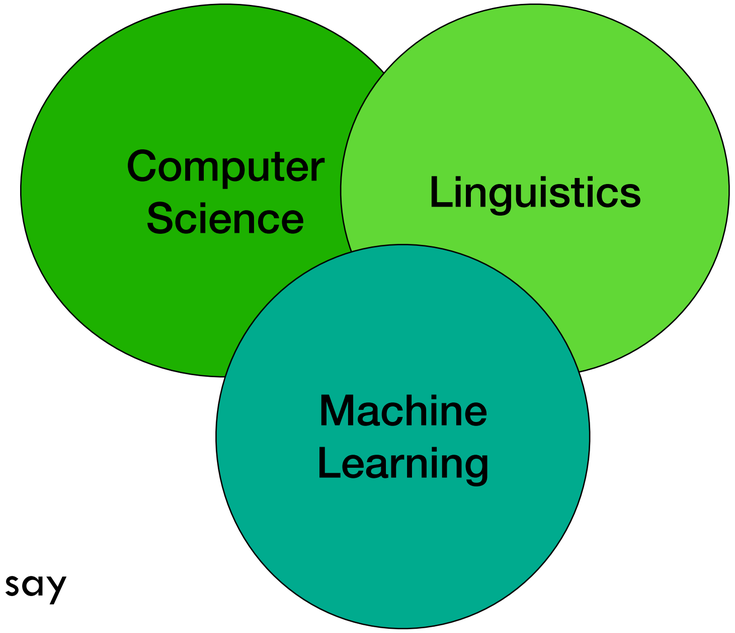
சங்கம்	கால இடைவெளி	கவிஞர்களின் எண்ணிக்கை	இராச்சியம் [9]	புத்தகங்கள் [9]
முதலில்	4440 ஆண்டுகள் [9]	549 [9]	பாண்டியா	புத்தகங்கள் எதுவும் பிழைக்கவில்லை
இரண்டாவது	3700 ஆண்டுகள் [9]	1700 [9]	பாண்டியா	*தொல்காப்பியம்* (ஆசிரியர் – தொல்காப்பியர்)
மூன்றாவது	1850 ஆண்டுகள் [9]		பாண்டியா	சங்க இலக்கியம் முழுவதையும் உள்ளடக்கியது



Ethnologue

Natural Language Processing

- Computer Science + Linguistics + ML
- Objective: Make computer understand language
- Broad categories of NLP:
  - Speech Recognition - Translation of speech to text
  - Natural Language Understanding - Computer's ability to understand what we say
  - Natural Language Generation - Make computers generate natural language



NLP: Techniques

Syntactic analysis (Syntax): Parsing the language with rules of formal grammar

Parse

```
(ROOT
  (S
    (NP
      (NP (NNP Surgical) (NN resection) (NNS specimens))
      (PP (IN of)
        (NP
          (NP (CD 85) (JJ invasive) (JJ ductal) (NNS carcinomas))
          (PP (IN of)
            (NP
              (NP (CD 85) (NNS women))
              (SBAR
                (WHNP (WP who))
                (S
                  (VP (VBD had)
                    (VP (VBN undergone)
                      (NP (CD 3D) (NN ultrasound))))))))))
    (VP (VBD were)
      (VP (VBN included)))
    (. .)))
```

Semantic analysis (Semantics): Process of understanding the meaning and interpretation of words, signs and sentence structure

- Synonymy: fall & autumn
- Hypernymy & hyponymy (is a): animal & dog
- Meronymy (part of): finger & hand
- Homonymy: fall (verb & season)
- Antonymy: big & small

Sentences that are syntactically correct need not be semantically correct

NLP: Data

As with the total number of languages, this count changes constantly. A language becomes endangered when its users begin to teach and speak a more dominant language to the children in the community. Due to their nature, endangered languages often have few speakers left, and it may be difficult to get information about them. Other times, the last known speaker of a language may die without public records.

As with the total number of languages, this count changes constantly.

A language becomes endangered when its users begin to teach and speak a more dominant language to the children in the community.

Due to their nature, endangered languages often have few speakers left, and it may be difficult to get information about them.

Other times, the last known speaker of a language may die without public records.

As
With
The
Total
Number
Of
Languages
This
Count
Changes
Constantly.

Corpus **Documents** **Paragraphs** **Sentences** **Tokens**

NLP: Pipeline

Standardisation

- Preprocess texts to a common format using different techniques

Sentence: The Sun@ Rises iN tHE EaST1!

i. Case normalisation: lowercase

the sun@ rises in the east1!

ii. Punctuation removal

the sun@ rises in the east1

iii. Remove unwanted symbols

the sun rises in the east

iv. Stop word removal

sun rises in east

Tokenization

- Process of splitting the text into smaller units
- Engrams - unigrams, bigrams, etc.

Sentence:

The lion is the king of the jungle.

Unigram:

The, lion, is, the, king, of, the, jungle.

Bigram:

The lion, lion is, is the, the king, king of, of the, the jungle.

NLP: Pipeline

Normalization

- Process of converting token into its base form (morpheme)
- Token can have the structure,

<prefix> <morpheme> <suffix>

Anti social ist

Stemming

- Rule-based process that removes inflectional forms from a token (stem)
- Stem need not be a meaningful word,

Example: “His teams are not winning”
Stem: **“”hi”, “team”, “are”, “not”, “winn”**

Lemmatization

- Step-by-step process of removing inflectional forms from a token using vocabulary, word structure, part of speech tags, and grammar relations (lemma)
- Lemmas are root words

Example: Running, Run, Ran >> Run

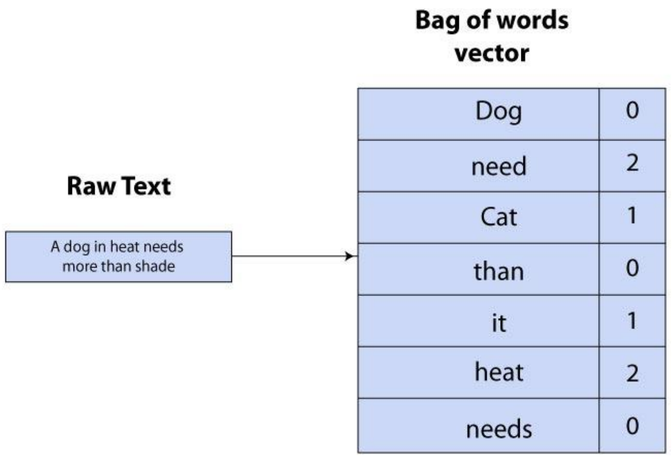
NLP: Pipeline

Vectorization

Maps words or phrases from vocabulary to a corresponding vector of real numbers (semantics)

Methods:

- Bag of words (BoW)
- Tf-idf (Term Frequency – Inverse Document Frequency)
- Word embeddings (Word2Vec)



Bag of words (BoW)

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

TF-IDF

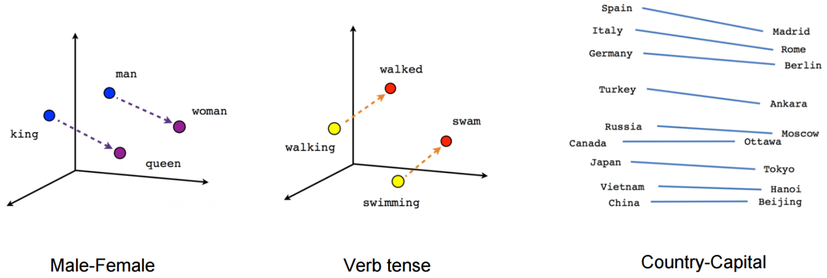
Term x within document y

$tf_{x,y}$ = frequency of x in y

df_x = number of documents containing x

N = total number of documents

TF-IDF



Male-Female | Verb tense | Country-Capital

Word embeddings

AI Tools using NLP

Easy (mostly solved)	Intermediate (good progress)	Hard (still hard)
Spell and Grammar checking	Information retrieval	Question answering
Text categorization tasks	Sentiment analysis	Summarization
Named-entity recognition tasks	Machine translation	Dialogue system
	Information extraction	

NLP for Indic Langauges

iNLTK

- Tokenization
- Word embeddings
- Sentence similarity
- Text completion

Indic NLP Library

- Normalization
- Transliteration
- Phonetic analysis
- Syllabification

StanfordNLP/ Stanza

- Lemmatization
- Parts-of-Speech (POS)
- Named Entity Recognition (NER)
- Dependency parsing

All these libraries support many Indian languages including Tamil but the quality of the output for Tamil still needs to be improved.

Open Research Challenges

What to build?

- Fundamental components like,
  - Tokenizer
  - Lemmatizer
  - Stemmer
  - Dependency Parsers - Shallow/Deep
  - POS Tagger
  - NER
  - Tree banks
  - Universal stop word list

Who should be involved?

- Linguists
- NLP experts
- ML researchers
- Research engineers

How to build?

- Leverage existing libraries and improvise them based on the grammar rules.
- Start from scratch by collecting relevant data and build the components ground-up.

Data Curation

Methods

• Manually curate high quality datasets depending on the task.

• Scrape content from books, websites, forums, Project Madurai etc.

• Validate the scraped or machine-generated data using linguists, and NLP experts.

• Validate existing multi-lingual datasets that contains Tamil and perform quality check (AI4Bharat, Bhashini, Aya by Cohere for AI, AI Tamil Nadu)

Challenges

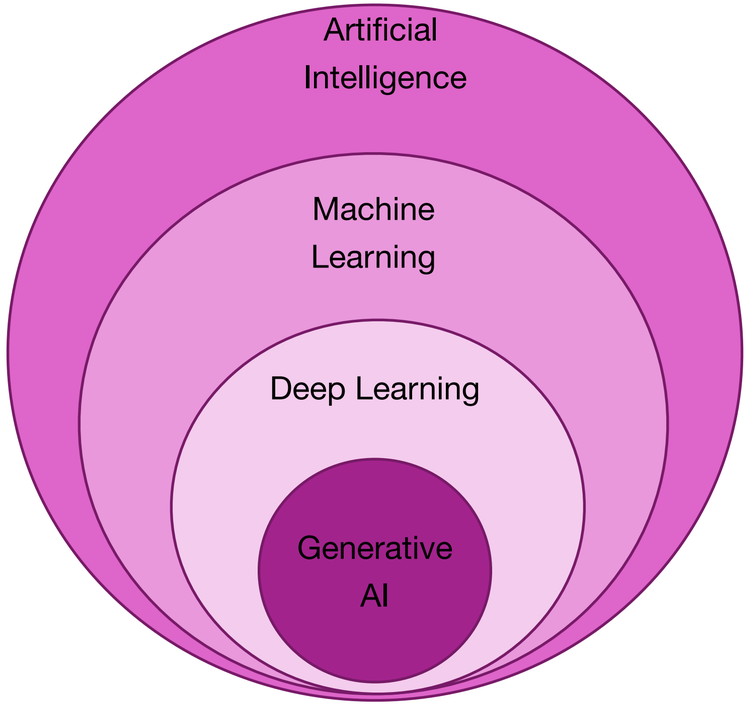
• Time consuming and labor-intensive.

• Identifying the right incentives for contributors.

• Consumer vs builder mindset shift.

• Stop looking down on the data curation process.

Types of AI



- **AI:** Build intelligent agents that can act like humans autonomously.

- **ML:** A machine learns the patterns in the data by training a model.

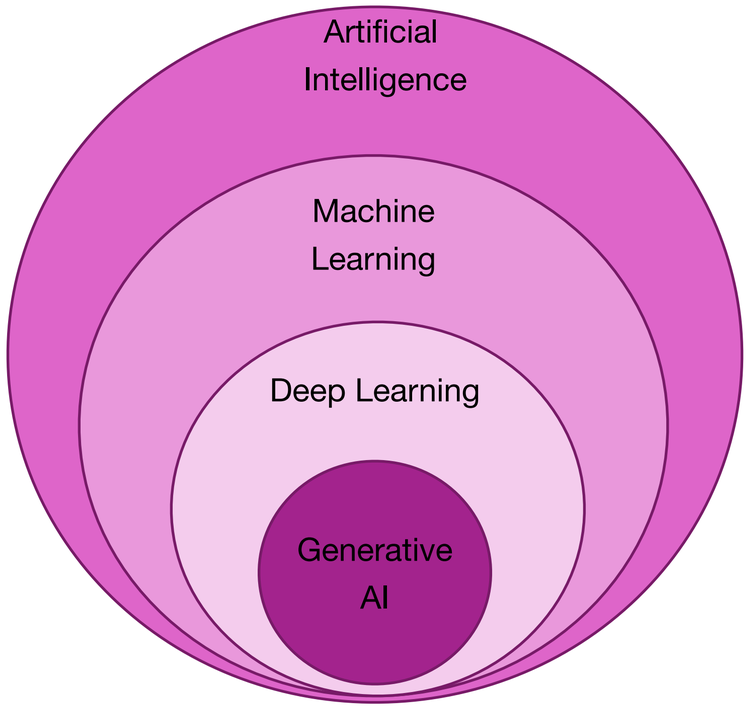
  - **Supervised learning** – Use labeled data, train models, predict on unseen data. Classification/Regression.

  - **Unsupervised learning** – Use unlabelled data to identify groups or clusters.

  - **Semi supervised learning** – uses little labeled data and more unlabelled data to train models.

  - **Reinforcement Learning** – An agent performs actions based on the environment and learns through trial and error (either rewarded or punished).

Types of AI



• **DL:** A neural network with interconnected nodes and layers is trained to learn complex patterns in the data. Uses the following methods of learning.

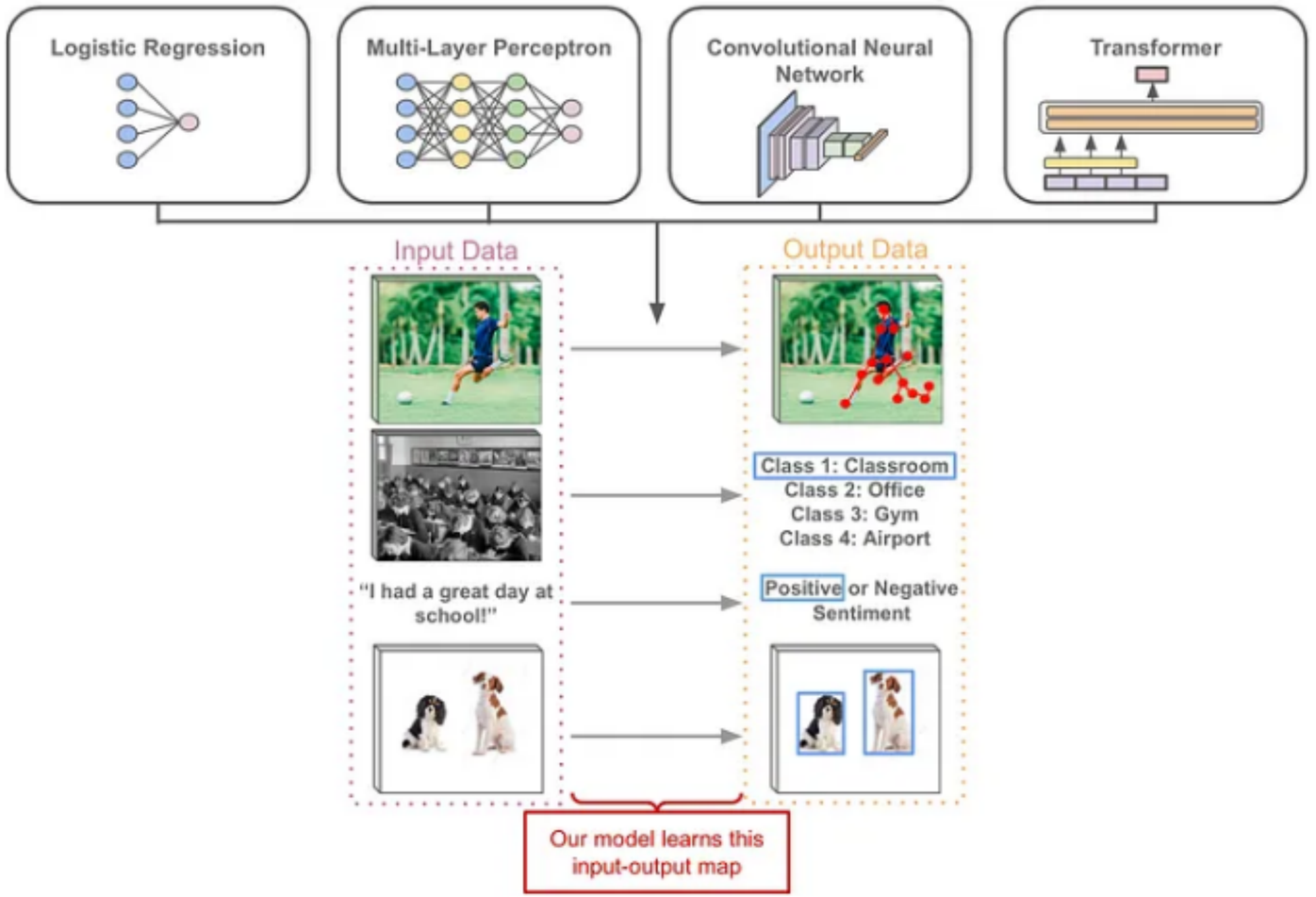
1. Supervised,
2. Unsupervised, and
3. Semi supervised

• **Generative AI:** It is a type of AI that can create new content, such as text, images, audio, and video.

- Learns from existing data
- Uses existing knowledge to generate new and unique outputs

https://www.cloudskillsboost.google/course_sessions/3016514/video/377870

Traditional Approach



https://towardsdatascience.com/language-models-gpt-and-gpt-2-8bdb9867c50a

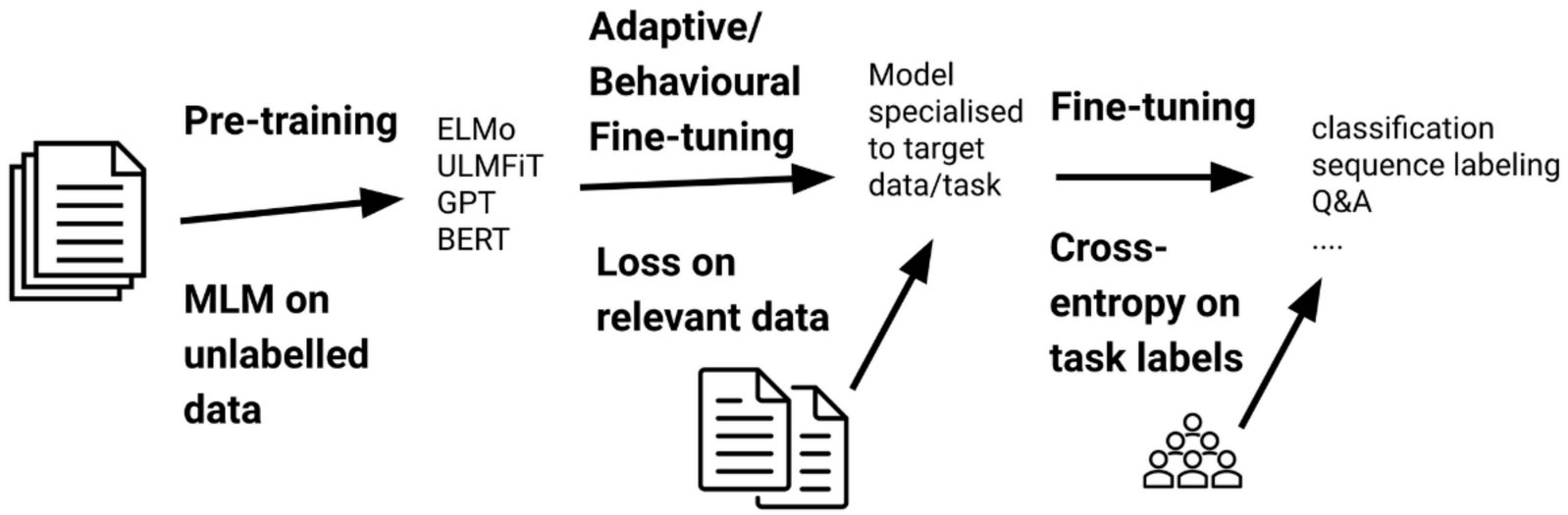
Transfer Learning

Step 1: Pre-training

- Use large amounts of generic data and train on a specific objective function.
- Unlabelled data is used to train on the language modelling objective like MLM.

Step 2: Fine-tuning

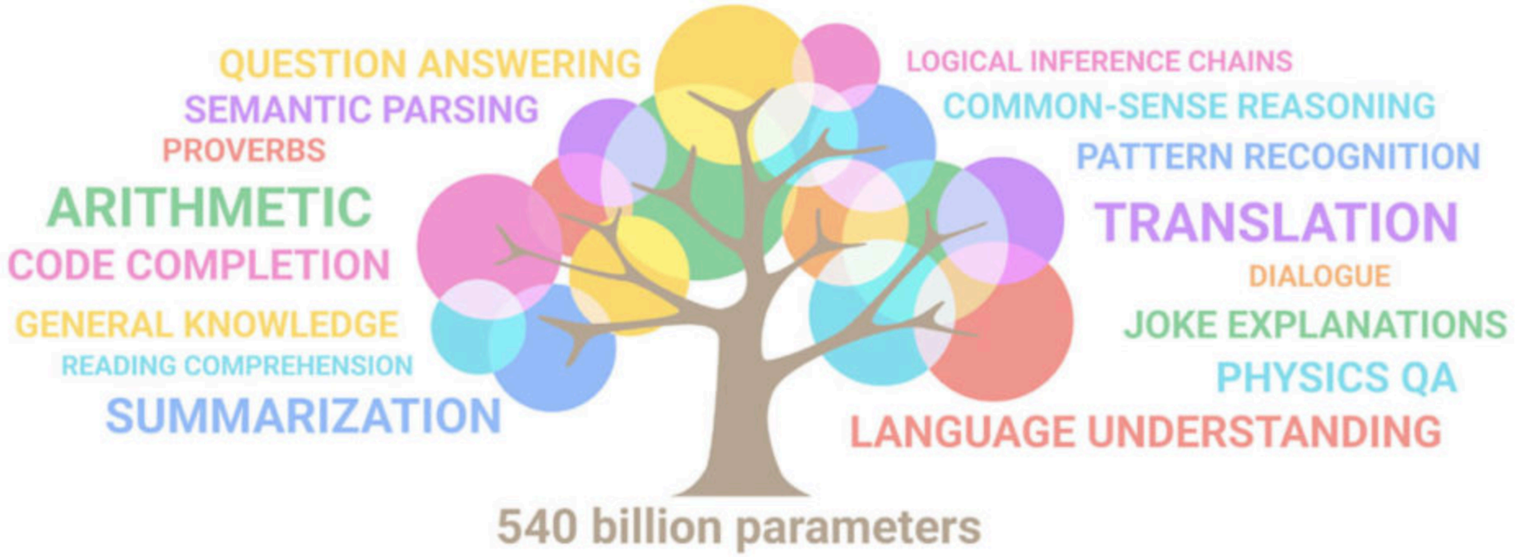
- Fine-tuning is done using task-specific objective function.
- Labelled data is used to fine-tune model on the downstream tasks.



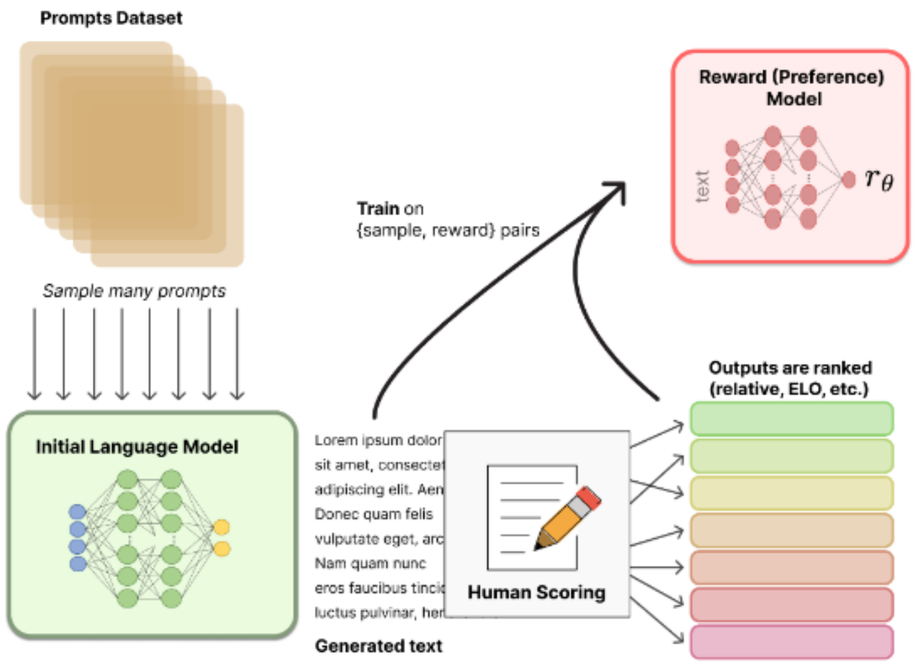
https://www.ruder.io/recent-advances-lm-fine-tuning/

Foundation Models / Large Language Models

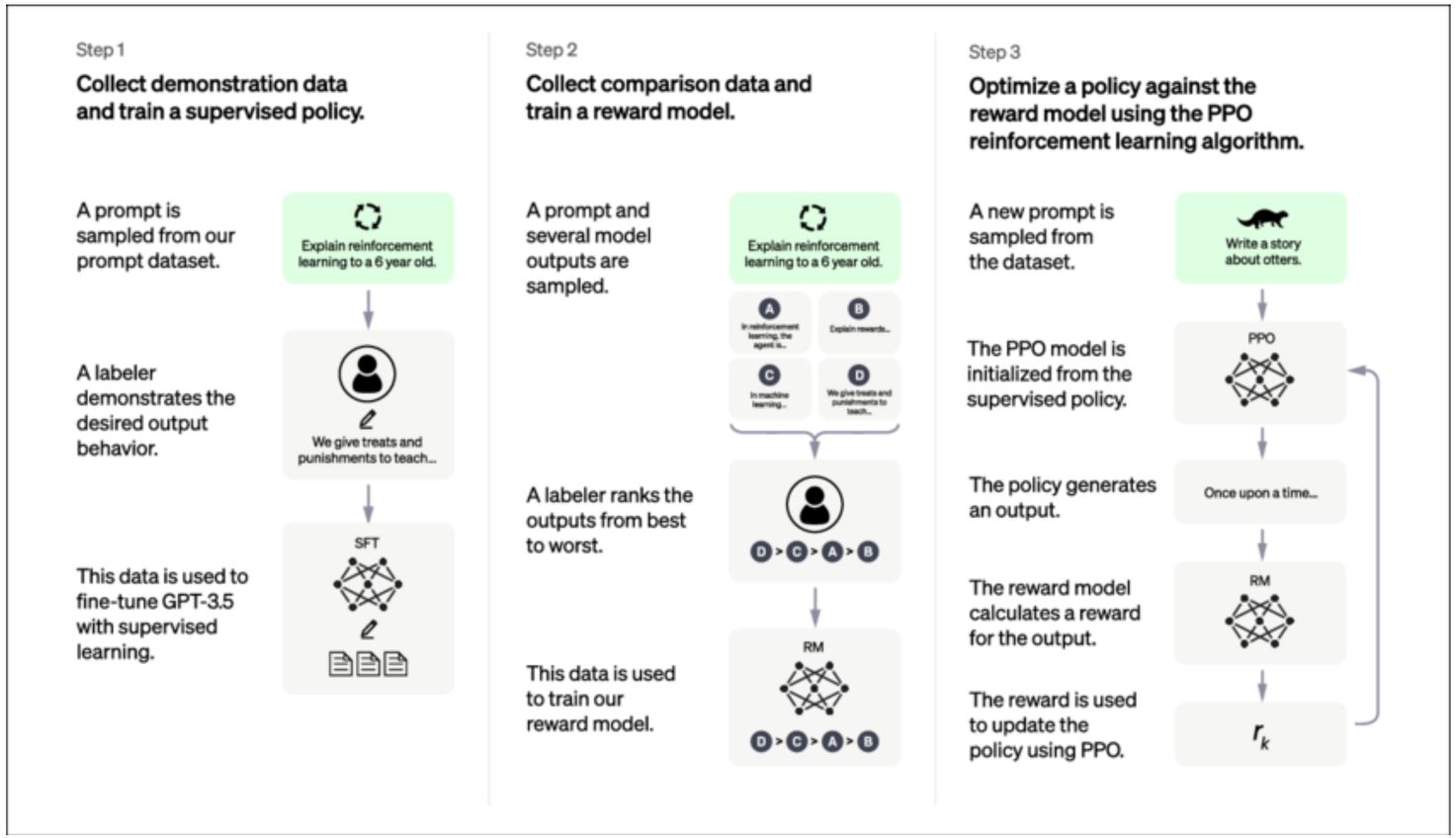
A foundation model is a large AI model pre-trained on a vast quantity of unlabelled data that was "designed to be adapted" (or fine-tuned) to a wide range of downstream tasks, such as sentiment analysis, image captioning, and object recognition. Prompt Engineering is used to interact with the model.



LLM Training: RLHF



https://huggingface.co/blog/rlhf



The ChatGPT training process. The figure is from OpenAI (2022a).

Prompt Engineering

- A prompt is a short piece of text that is given to the large language model as input, and it can be used to control the output of the model in many ways.

- Designing this prompt efficiently is called prompt engineering.

- Methods
  - Zero-shot
  - Few-shot
  - Chain of Thought

A prompt contains any of the following elements:

Instruction - a specific task or instruction you want the model to perform

Context - external information or additional context that can steer the model to better response

Input Data - the input or question that we are interested to find a response for

Output Indicator - the type or format of the output.

https://www.promptingguide.ai/introduction/elements

Limitations of Generative AI

- Hallucinations are words or phrases that are generated by the model that are often nonsensical or grammatically and factually incorrect.
  - The model is not trained on enough data. Misleading information.
  - The model is trained on noisy or dirty data. Garbage in => Garbage out!
  - The model is not given enough context. Incomplete information.
  - The model is not given enough constraints. Anyone can use it.

- Ethical concerns – what if the models are biased and are used for unintended purpose.

- Productionizing the LLMs is difficult.
  - Cost — Infrastructure
  - Time — Takes longer to build your own LLMs. Pre-training vs Fine-tuning vs Prompting.

- Explainability is difficult.

Gen AI: Research Directions

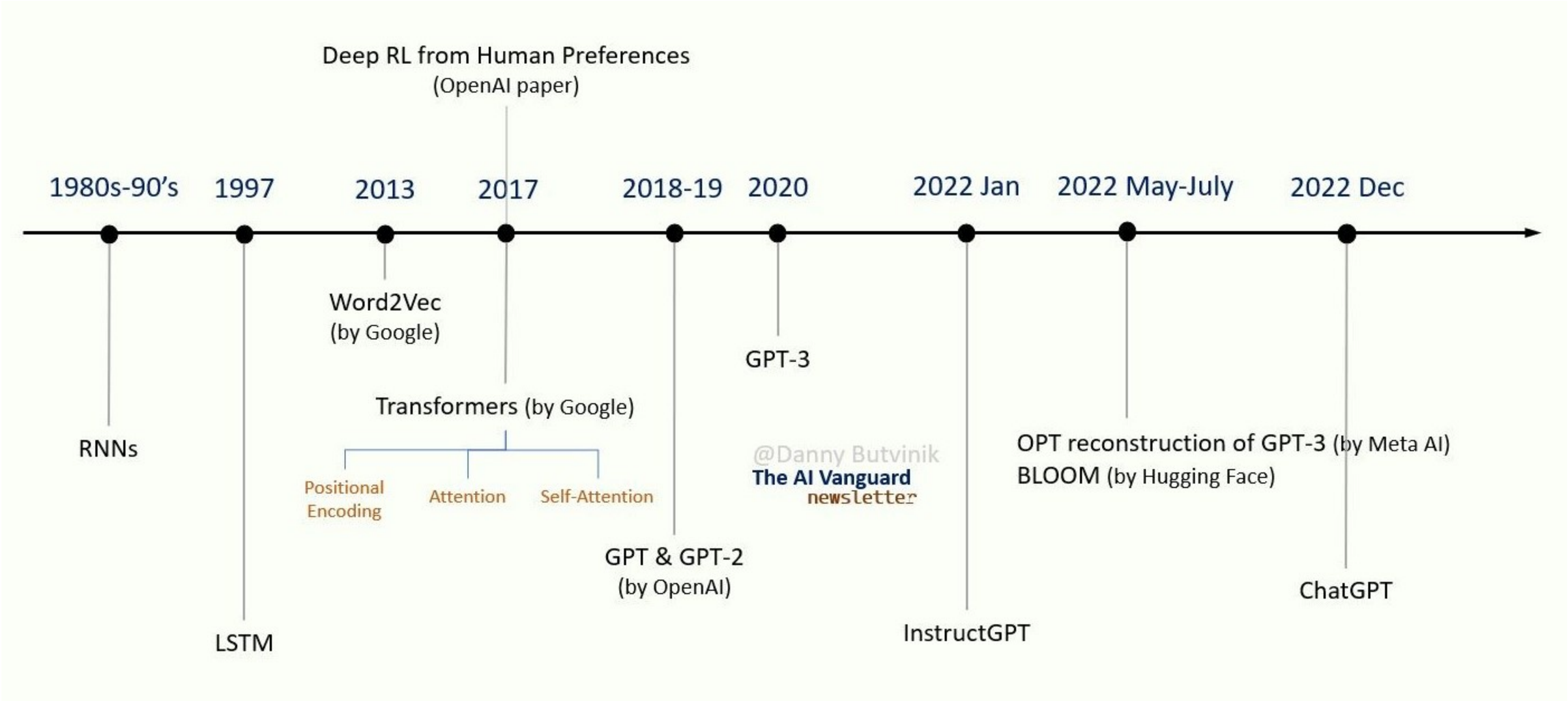
Open challenges in LLM research

1. Reduce and measure hallucinations
2. Optimize context length and context construction
3. Incorporate other data modalities
4. Make LLMs faster and cheaper
5. Design a new model architecture
6. Develop GPU alternatives
7. Make agents usable
8. Improve learning from human preference
9. Improve the efficiency of the chat interface
10. Build LLMs for non-English languages

Hardest is to build LLMs for non-English languages!

https://huyenchip.com/2023/08/16/llm-research-open-challenges.html

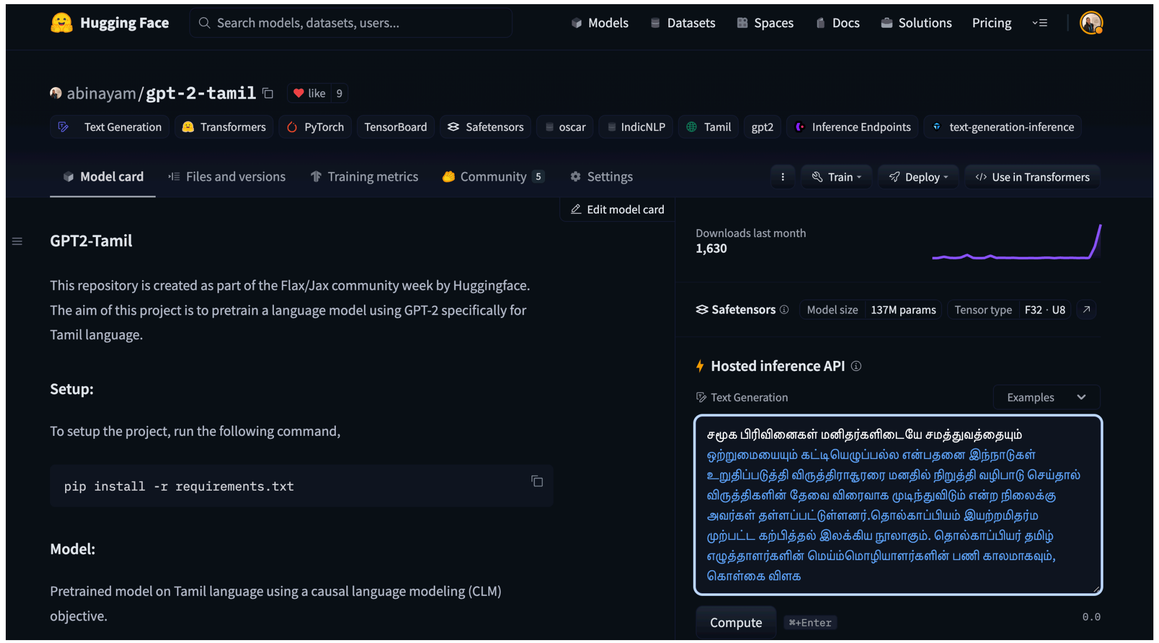
NLP Timeline



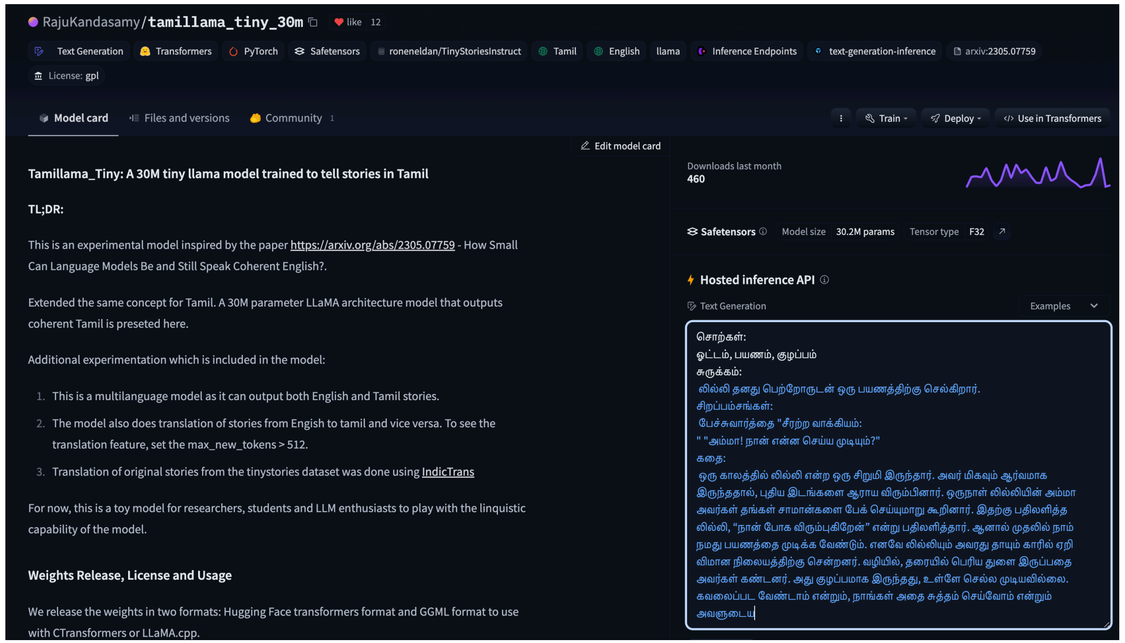
Generative AI for Tamil

Tamil is a culturally rich language with a great wealth of information spanning across literature, music and the fine arts!

GPT-2 Tamil (Abinaya Mahendiran)



Tamillama (Raju Kandasamy)



Role of the Community

What does it take to build a Tamil Generative AI?

- Building hyperlocal community and educate about AI - AI Tamil Nadu (some chapters have been doing it for many years).
- Foster open source projects that digitises Tamil literature like Project Madurai, and initiatives like Bhashini, AI4Bharat, and Aya by Cohere for AI aimed at curating high-quality data and building multi-lingual models.
- Managing infrastructure cost through crowd-sourcing or CSR activities.
- Motivating and teaching people to contribute high quality data to solve for specific problems faced by the Tamil community.
- Imparting technical knowledge (NLP) and do fundamental research for Tamil computing.
- Put up regulations to handle the ethical and societal issues (involve government).

Thank You :)

Questions?