Large Language Model for Telugu

Birudugadda Srivibhav (22110050), Pavan Deekshith (22110190)
Computer Science and Engineering, Indian Institute of Technology Gandhinagar

Project Supervisor: Professor Mayank Singh, Computer Science And Engineering
The report was submitted on 14-07-2024

Abstract: *This report chronicles the progress made from May 4, 2024, to July 9, 2024, towards developing a Telugu Language Model (LLM). The project aims to address the scarcity of comprehensive language models tailored for the Telugu language, focusing on understanding existing models, collecting diverse datasets, and preparing the groundwork for model development.*

Problem Statement

The project aims to develop a Telugu Language Model (TLM) to enhance natural language processing tasks in Telugu. Despite a growing demand for language technologies in Indic languages like Telugu, existing models lack depth and sophistication, limiting their effectiveness. This project seeks to address these limitations by designing a more robust TLM capable of understanding and generating Telugu text.

Contributions

In this period, significant contributions have been made:

1) **Exploration of Existing Models:**
   Investigated and analysed several existing Telugu Language Models (LLMs) such as Chandamama Kathalu, Llama-3-8b-Telugu_Romanized, POS-NER models, Comprehension QA models, Summarization Models, Whisper-large-v3, and Navarasa 2.0. Each model's capabilities, datasets used, and applicability to Telugu language tasks were reviewed.

2) **Dataset Curation:**
   Curated a diverse array of Telugu text datasets from sources including historical archives (e.g., Eenadu from Wayback Machine, Andhra Jyothi), contemporary news sites (Telugu360, sakshi.com), pdfs (school textbooks), and public datasets (Roots Corpus, AI4Bharat-IndicNLP). Efforts focused on collecting large-scale, domain-specific datasets to ensure comprehensive coverage.

3) **Methodological Development:**
   Developed robust methodologies for web scraping, PDF-to-text conversion, and deduplication to enhance dataset collection efforts. We addressed challenges in handling complex PDF structures and ensuring accurate text extraction from scanned documents. Additionally, we undertook the significant task of developing deduplication algorithms from scratch, aiming to efficiently remove redundant data and ensure dataset cleanliness.

Prior Work

Prior research provided foundational insights into the landscape of Telugu Language Models and datasets. Existing models were assessed for their strengths in various linguistic tasks, highlighting gaps in coverage and usability across different domains.

Major Limitations of Prior Work

1) **Multilingual Models Not Specifically for Telugu:** Many models developed for language processing are multilingual, designed to handle a wide range of languages rather than being specifically optimised for Telugu. While this approach enables the models to process multiple languages, it often results in suboptimal performance for individual languages like Telugu. This limitation highlights the need for developing models specifically tailored for Telugu, ensuring they are trained on comprehensive Telugu datasets to capture the language's intricacies effectively.

2) **Lack of Diverse Datasets:** The effectiveness of language models heavily depends on the quality and diversity of the training datasets. Unfortunately, many datasets used to train Telugu language models lack diversity, often comprising texts from limited domains such as only news articles. Without exposure to a wide range of linguistic styles, registers, and contexts, models may fail to understand and generate text that accurately reflects the real-world use of Telugu.

3) **Task-Specific Limitations:** Many Telugu language models are developed with a focus on specific tasks, such as translation, sentiment analysis, or named entity recognition. While these specialised models can perform well in their designated areas, they often lack the versatility required for general natural language processing tasks.

Methodology

The methodology outlined here describes the systematic approach taken to develop a comprehensive Telugu Language Model (LLM) from May 4, 2024, to July 9, 2024. The process involved several key stages: researching existing Telugu LLMs, collecting and processing diverse datasets, interacting with models, developing data deduplication techniques, data cleaning and tokenisation.

Stage 1: Research and Analysis of Existing Telugu LLMs
Objective: To understand the current landscape of Telugu Language Models and identify their capabilities, datasets used, and potential applications.

1) **Identification of Models:** Conducted an extensive search for existing Telugu LLMs, including Chandamama Kathalu, Llama-3-8b-Telugu_Romanized, POS-NER models, Comprehension Question Answering models, Summarization Models, Whisper-large-v3 by OpenAI, and Navarasa 2.0.

2) **Analysis of Capabilities:** Evaluated each model's strengths and weaknesses, focusing on their datasets, training methodologies, and specific NLP tasks they address.

Stage 2: Data Collection and Dataset Analysis
Objective: To gather a comprehensive and diverse set of Telugu text data to support the development of a robust LLM.

1) **Identification of Existing Datasets:** Compiled a list of existing Telugu datasets, including Roots Corpus, AI4Bharat-IndicNLP Dataset, CC-100, MC4, OSCAR-2109, OSCAR-2201, OSCAR-2301, and Kaggle/Telugu_NEWS.

2) **Web Scraping for Additional Data:** Developed systems to scrape text data from various websites and archives, including Eenadu from the Wayback Machine, Telugu360 News, and sakshi.com.

3) **PDF to Text Conversion:** Utilised OCR technologies to extract text from PDF files, supplementing web-scraped data.

Stage 3: Data Deduplication
Objective: To clean and preprocess the collected data, ensuring its uniqueness and readiness for model training.

1) **Data Segmentation:** Divided the 150 GB of collected text data into smaller files, each containing 800-1000 words, to facilitate efficient deduplication. The idea is to keep only one text file and drop all its neighbours.

2) **Calculating Sim-Hashes:** After dividing the data into smaller text files, we have around 4.5 crore files. We then calculate the sim-hash for each text file and store them in CSV files. The original idea was to perform a pair-wise comparison of hashes across all CSV files (later identified as computationally expensive and inefficient).

3) **Transition to Min-Hash Algorithm:** Due to the inefficiency of the sim-hash approach, we transitioned to using the min-hash algorithm to identify and remove similar documents more efficiently. The min-hash algorithm, implemented through the datasketch library's Locality Sensitive Hashing (LSH), allows us to calculate the nearest neighbours of each document. This method is significantly more efficient than pairwise comparisons, which were computationally infeasible for our large dataset. By using min-hash LSH, we can effectively detect and eliminate duplicate and near-duplicate documents, ensuring a more streamlined and accurate deduplication process.

Stage 4: Data Cleaning
Objective: To ensure the data is free from vulgar content, personal information, excessive English text, and promotional material.

1) **Identification of Unwanted Content:** Developed scripts to identify and list rows containing vulgar words, contacts, and personal information.

2) **Separation of Data:** Utilized the identified list to separate the dataset into folders containing 'good' data (clean and usable) and 'bad' data (contaminated with unwanted content).

3) **Removal of Dates and Promotions:** Implemented additional scripts to remove promotions and ads from the dataset, logging detected promotions and replacing links with placeholder tokens.

4) **Filtering Excessive English Text:** Applied a threshold to filter out rows with excessive English words, ensuring the dataset remained predominantly Telugu.

Stage 5: Tokenization and Calculation of Fertility Score
Objective: To prepare the dataset for model training by segmenting text into subword units and calculating fertility scores.

1) **Dataset Preparation:** Created five batches for tokeniser training from a 15% (38GB) partition of a deduplicated dataset totalling 235GB. The batches are as follows: Batch-1 (4.1 GB), Batch-2 (9.9 GB), Batch-3 (15 GB), Batch-4 (25 GB), and Batch-5 (38 GB). The first four batches were randomly sampled subsets from the 15% partition, while Batch-5 encompassed the entire 38GB of data.

2) **Subword Segmentation with SentencePiece BPE:** Utilized the SentencePiece BPE Tokenizer to segment text into subword units, creating a 32,768 token vocabulary. This approach, combining SentencePiece and Byte-Pair Encoding (BPE), was especially effective for agglutinative languages like Telugu, enhancing overall language modelling performance.

A detailed table showing the experiments performed:
Vocab size - 32768 fixed for all experiments.

S.No	1000 Sentences		5380 Sentences	
	Average	Maximum	Average	Maximum
Batch_1(4.1GB)	1.7175	5.066	1.9044	11.22
Batch_2(9.9GB)	2.784	5.5	2.891	12.33
Batch_3(15 GB)	2.784	5.5	2.891	12.33
Batch_4(25 GB)	2.784	5.5	2.891	12.33
Batch_5(38 GB)	2.784	5.5	2.891	12.33

S.N0	Frequency = 5	Frequency = 7	Frequency = 9
Batch_1	1.7175	1.7175	1.7175

Conclusion

The development of the Telugu Language Model has progressed significantly from initial research to substantial data collection and deduplication. Despite facing challenges, such as the inefficiency of sim-hash for large datasets, transitioning to the min-hash algorithm has allowed us to handle deduplication more effectively. Moving forward, the next critical steps will involve pre-training the language model on the curated dataset, followed by fine-tuning it on specific downstream tasks. Pre-training will help the model learn the general structure and nuances of the Telugu language, while fine-tuning will adapt it for particular applications such as sentiment analysis, question answering, and text summarisation. By implementing these stages, we aim to create a comprehensive and accurate language model that can significantly contribute to various NLP tasks in the Telugu language, supporting both academic and practical applications.

References

- H. Laurençon *et al.*, "The BigScience ROOTS Corpus: a 1.6TB composite multilingual dataset," *arXiv.org*, Mar. 07, 2023. https://arxiv.org/abs/2303.03915
- R. Ángel, "Dataset deduplication using spark's MLlib - Towards Data Science," *Medium*, Dec. 08, 2021. [Online]. Available: https://towardsdatascience.com/deduplication-using-sparks-mllib-4a08f65e5ab9
- ChenghaoMou, "GitHub - ChenghaoMou/text-dedup: All-in-one text de-duplication," *GitHub*. https://github.com/ChenghaoMou/text-dedup/tree/main

Acknowledgements

I extend my sincere gratitude to Prof. Mayank Singh for his continuous support, guidance, and encouragement throughout the project. I would also like to acknowledge M.Tech students Hitesh Lodwal and Aamod Thakur for their valuable contributions and assistance, which was pivotal in shaping the early stages of this ongoing endeavour.