Leveraging RoBERTa for Multilingual Question and Answering

Project report of
Year-V Semester-X

By

Yash Oza	**C065**
Rushank Shah	**C081**
Shubham Sheth	**C099**
Arhya Singh	**C104**

Prof. Khushbu Chauhan

Department of Computer Engineering

Mukesh Patel School of Technology Management & Engineering

NMIMS (Deemed-to-be University), Mumbai

INDEX OF CONTENT

INDEX OF FIGURE

INDEX OF TABLE

1. Introduction

Given that India's population is rapidly approaching 1.4 billion, it is imperative that the underrepresentation of Indian languages on the internet, including Hindi and Tamil, be addressed. Because of this linguistic variation, Natural Language Understanding (NLU) models face specific difficulties, especially when it comes to correctly predicting questions' answers—a crucial NLU job. The state of multilingual modeling is not advanced enough to handle Indian languages efficiently, which results in less-than-ideal user experiences for Indian users in downstream web applications.

We introduce a new method to tackle this urgent problem by utilizing chaii-1, a carefully selected question-answer dataset that consists of native-language Hindi and Tamil question-answer pairings. Our dataset has been painstakingly created by skilled data annotators, as opposed to earlier attempts that relied on translation, guaranteeing the authenticity and accuracy of the linguistic subtleties inherent in Indian languages.

Our main goals are to improve the online experience for the large Indian population while also advancing the field of multilingual natural language processing. We hope to stimulate the creation of more high-quality datasets and the use of chaii-1 to accelerate the creation of reliable NLU models for Indian languages. Furthermore, our contributions go beyond the linguistic landscape of India, providing methods and insights that are transferable to other languages that are underrepresented worldwide.

We use the latest RoBERTa model architecture to generate predictions and show how well it handles Indian languages. We demonstrate the effectiveness of our strategy in bridging the language gap and promoting a more inclusive digital ecosystem for Indian users through thorough experimentation and evaluation. Through the provision of precise question answering services in Tamil and Hindi, we create the foundation for revolutionary developments in information retrieval, web accessibility, and multilingual natural language processing, enabling millions of people to interact and navigate with online content in their mother tongues.

2. Literature survey

The Google search is powered by the encoder-only BERT [1] architecture. Having been trained on several tasks, BERT has a multi-task purpose. It can handle lengthy input contexts and has been trained on the whole Wikipedia corpus. Both sentence and token levels of training were used. Using the masking technique, BERT makes a prediction for the next sentence. Tasks like Question and Answering and Single Sentence Classification can benefit from the application of BERT.

The robustly optimized Bert method, or RoBERTa [2], employs dynamic token masking. Thus, randomly masking the tokens for several epochs is the main idea. Additionally, since Next Sequence Prediction was not very helpful, it is ignored. The tokenization procedure and batch sizes are where BERT and RoBERTa diverge. Additionally, they expanded the dataset's size.

The goal of extractive question answering (EQA) is to retrieve text segments from a particular context in order to provide answers. A single-span EQA or a multi-span EQA can be distinguished by the quantity of spans that are extracted. - Early research on single-span EQA captured the interplay between questions and settings using attention mechanisms. Models such as QANet and BIDAF enhanced this. To overcome this, two new pre-training techniques were introduced: prompt-tuning and recurrent span selection. The research of multi-span EQA was made possible by databases such as DROP. Various answer kinds were handled by models such as NAQANet and NABERT+. In conclusion, the survey examined how EQA procedures have evolved from single to multi-span jobs and emphasized important strategies that have been put forth as well as unresolved issues.

To answer questions, machine reading comprehension (MRC) entails comprehending and extracting information from a given context. Research in machine learning (MRC) is ongoing and crucial for applications such as conversational agents and question answering systems. - For MRC tasks, a variety of deep learning methods, including CNN, RNN, LSTM, and pre-trained language models like BERT, have been applied. Models based on BERT have demonstrated strong performance on several NLP tasks, such as MRC. - Important multilingual BERT models that have been researched for multilingual MRC include XLM- RoBERTa, RemBERT, and MURIL. MURIL is a language model primarily for Indian languages, whereas RemBERT and XLM-RoBERTa are multilingual models with broader applications.

3. Proposed Work

3.1 Dataset

The dataset is taken from Google-organized Kaggle competition "chaii — Hindi and Tamil Question Answering." Columns including id, context, question, answer_text, answer_start, and language are included in the dataset. Finding the solutions to the questions posed in sections written in Indian languages is the goal. There are 368 examples in Tamil and 747 examples in Hindi in the training set. There are five examples in the test data, but the answer_text and answer_start details are missing.

3.2 RoBERTa -A Robustly Optimized BERT Pretraining Approach

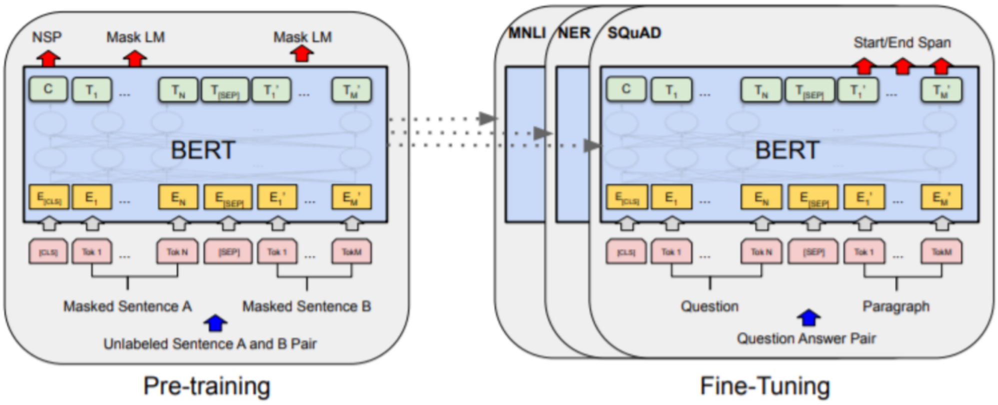


Figure 3.2.1: Architecture of BERT and RoBERTa model

The BERT model is evolved into RoBERTa, which improves performance and adaptability in tasks involving natural language processing by introducing substantial changes to the pre-training and fine-tuning processes. Although RoBERTa and BERT have identical architecture, RoBERTa optimizes important hyperparameters and makes small embedding adjustments to increase efficiency. In contrast to BERT, RoBERTa uses bigger mini-batches and higher learning rates during training instead of implementing the next-sentence pretraining target. Furthermore, RoBERTa uses a different pretraining approach and substitutes a character-level BPE vocabulary, similar to GPT-2, for the byte-level BPE tokenizer. Significantly, RoBERTa uses the separation token tokenizer.sep_token to ease segment separation, doing away with the requirement for token_type_ids. The combination of these improvements strengthens RoBERTa's ability to handle various downstream NLP jobs. RoBERTa's capacity to comprehend context and produce precise answers depending on the input question and passage makes it especially well-suited for question-answering (Q&A). RoBERTa can be trained on a particular dataset of question-answer pairs in Q&A applications to help it understand the connections between questions and responses. We use the start and the end spans to extract the answers from within the context, instead of generating an answer like LLMs.

4. Methodology

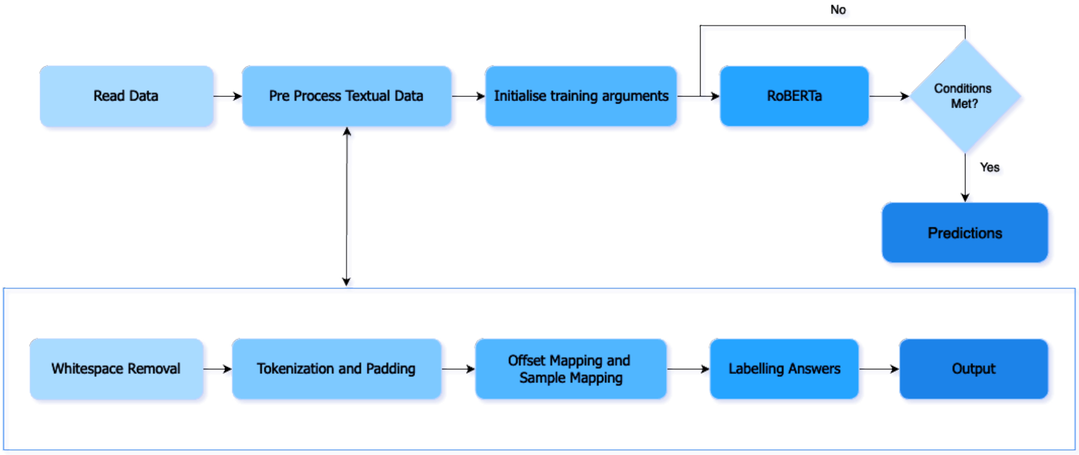


Figure 4.2: Project Pipeline

The methodology can be understood via the means of the pipeline above. We read and preprocess the textual data to make them streamline as inputs for our model. This preprocessing ensures the model computes and comprehends all training data.
The preprocessing pipeline includes:

Whitespace removal

We remove the whitespaces to prevent exponentially increased high dimension of input features as tokenizer considers whitespaces to be legitimate tokens. Keeping the whitespaces might make the tokens illegible and difficult to comprehend.

Tokenization & Padding

Tokenization is the process of breaking the context down by a certain number of characters. Model inputs have a constraint on the number of tokens hence we introduce a stride. Stride helps us handle overflowing tokens and thus prevents loss of important context. To ensure uniform length of features, we use padding. Padding makes all the features the same dimension based on the size of the biggest feature.

Offset Mapping & Sample Mapping

Sample mapping helps us keep track of which features belong to the same original example & offset mapping helps us understand where each token originated in the original text. During the training and assessment phases of the question-answering model, these mappings are essential for accurately determining the beginning and ending locations of replies.

Labelling Answers

Answers are labelled based on the start and end positions.
The start and finish positions are set to the CLS token's index if no responses are received. It calculates the start and end character indices of the answer in the text for examples that have answers. To find the beginning and ending token positions of the response, it then transforms these character indices to token indices. The feature is labelled with the CLS token index if the response is outside of the span. If not, the start and finish positions are set to the token indices of the solution.

After preprocessing we initialize training arguments. We employ TrainingArguments, which is a class provided by the Hugging Face transformers library, which allows us to specify various parameters related to the training process. Parameters such as evaluation_strategy, save_strategy, learning_rate, batch_size, num_train_epochs, warmup_ratio, gradient_accumulation_steps, and weight_decay are defined here.
These parameters control how the training will be conducted, including how often evaluation is performed, how often model checkpoints are saved, the learning rate schedule, and other optimization settings.

The entire training procedure is coordinated by the initialization of the Trainer class from the transformers library.

Parameters are

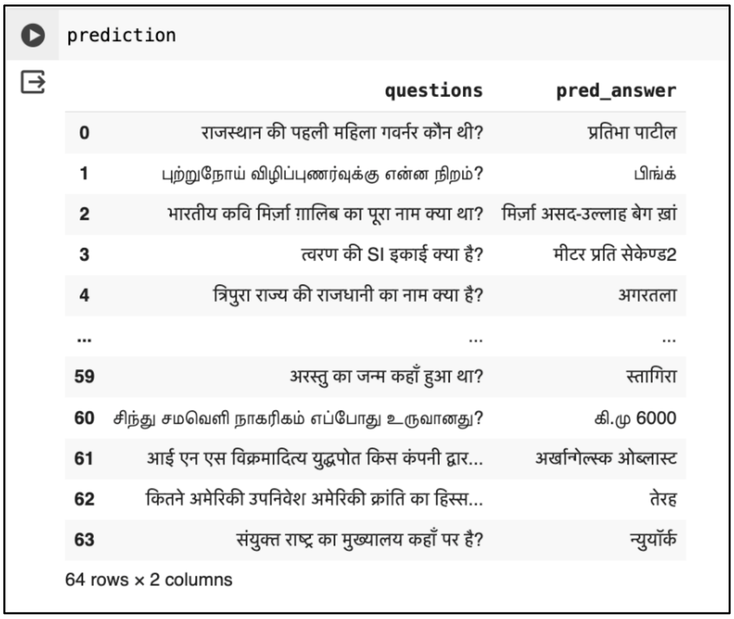
- **model**: The model that we wish to train to respond to questions.
- **args**: Training-related parameters are contained in the TrainingArguments object. The datasets with training and validation examples are called **train_dataset** and **eval_dataset**, respectively.
- **data_collator**: The object or method in charge of gathering training batch data.
- **tokenizer**: The input data is tokenized using a tokenizer.

The Trainer is set up to run the training loop, assess the model's performance on the validation set, and save model checkpoints according to the selected strategies by passing these inputs.
The model learns from the training data iteratively during training, modifying its parameters (weights) to minimize the loss function.

The model's performance is assessed on the validation set at the conclusion of each epoch using the evaluation technique that was laid out in the training arguments. The training arguments also define how checkpoints are saved.
After training, we make predictions by loading our saved model. These raw predictions require post-processing to ensure human readability.

5. Experimental results and analysis

After fine-tuning, we obtained a loss of 0.275.
There is no proper metric to define our model's performance except the training loss and the results obtained after using the saved model.



```
prediction
```

	questions	pred_answer
0	राजस्थान की पहली महिला गवर्नर कौन थी?	प्रतिभा पाटील
1	புற்றுநோய் விழிப்புணர்வுக்கு என்ன நிறம்?	பிங்க்
2	भारतीय कवि मिर्ज़ा ग़ालिब का पूरा नाम क्या था?	मिर्ज़ा असद-उल्लाह बेग ख़ां
3	त्वरण की SI इकाई क्या है?	मीटर प्रति सेकेण्ड2
4	त्रिपुरा राज्य की राजधानी का नाम क्या है?	अगरतला
...	...	...
59	अरस्तु का जन्म कहाँ हुआ था?	स्तागिरा
60	சிந்து சமவெளி நாகரிகம் எப்போது உருவானது?	கி.மு 6000
61	आई एन एस विक्रमादित्य युद्धपोत किस कंपनी द्वार...	अर्खान्गेल्स्क ओब्लास्ट
62	कितने अमेरिकी उपनिवेश अमेरिकी क्रांति का हिस्स...	तेरह
63	संयुक्त राष्ट्र का मुख्यालय कहाँ पर है?	न्युयॉर्क

64 rows × 2 columns

Figure 5.3: Training Results

Table 5.1: Custom Inputs

id	context	question	language	PredictionString
1	निफ्टी या निफ्टी50, नेशनल स्टॉक एक्सचेंज (एनएसई) का एक प्रमुख सूचकांक है, जो 13 प्रमुख क्षेत्रों से एनएसई पर कारोबार करने वाली शीर्ष 50 कंपनियों का प्रतिनिधित्व करता है। यह सूचकांक निवेशकों को बाजार की भावनाओं और प्रदर्शन पर एक विहंगम दृष्टि देता है। इसलिए, इसे बड़े पैमाने पर भारतीय शेयर बाजार और अर्थव्यवस्था का सच्चा प्रतिबिंब माना जाता है। इसके कारण, कई लोग इस सूचकांक के प्रदर्शन को एक अवधि में अपने पोर्टफोलियो के प्रदर्शन के मुकाबले एक बेंचमार्क मानते हैं। इसे 22 अप्रैल 1996 को लॉन्च किया गया था, यानी इस साल यह 25 साल का हो गया! 22 अप्रैल, 1996 को 1,107 से 15 जून 2021 को 15,901.60 की नई रिकॉर्ड ऊंचाई तक इसने कितनी अद्भुत यात्रा देखी है।	निफ्टी50 क्या है?	hindi	नेशनल स्टॉक एक्सचेंज (एनएसई) का एक प्रमुख सूचकांक

2	राउज एवेन्यू कोर्ट द्वारा आम आदमी पार्टी (आप) के संयोजक को 15 अप्रैल तक न्यायिक हिरासत में भेजने का आदेश दिए जाने के बाद दिल्ली के मुख्यमंत्री अरविंद केजरीवाल को तिहाड़ जेल ले जाया गया। कई आप नेताओं ने तिहाड़ जेल के बाहर विरोध प्रदर्शन जारी रखा। दिल्ली के मुख्यमंत्री अरविंद केजरीवाल को न्यायिक हिरासत में भेजे जाने के बाद उनकी पत्नी सुनीता केजरीवाल ने केंद्र की आलोचना करते हुए कहा कि "देश की जनता इस तानाशाही का जवाब देगी।" सुनीता केजरीवाल ने पूछा, "अगर जांच पूरी हो गई थी तो उन्हें जेल क्यों भेजा गया? देश की जनता इस तानाशाही का जवाब देगी।" इससे पहले, अदालत ने 28 मार्च को केजरीवाल की ईडी हिरासत 1 अप्रैल तक बढ़ा दी थी। प्रवर्तन निदेशालय ने कथित शराब नीति मामले में 21 मार्च को आप सुप्रीमो को गिरफ्तार किया था।	अरविंद केजरीवाल कौन हैं?	hindi	दिल्ली के मुख्यमंत्री
3	राजा कृष्णदेव राय के राज्य में चेलाराम नाम का एक व्यक्ति रहता था। वह राज्य में इस बात से प्रसिद्ध था कि अगर कोई सुबह-सवेरे उसका चेहरा सबसे पहले देख ले तो उसे दिनभर खाने को कुछ नहीं मिलता। लोग उसे मनहूस कहकर पुकारते थे। बेचारा चेलाराम इस बात से दुखी तो होता, लेकिन फिर भी अपने काम में लगा रहता। एक दिन यह बात राजा के कानों तक जा पहुंची। राजा इस बात को सुनकर बहुत उत्सुक हुए। वह जानना चाहते थे कि क्या चेलाराम सच में इतना मनहूस है? अपनी इस उत्सुकता को दूर करने के लिए उन्होंने चेलाराम को महल में हाजिर होने का बुलावा भेजा। दूसरी ओर चेलारम इस बात से अंजान खुशी-खुशी महल के लिए चल पड़ा। महल पहुंचने पर जब राजा ने उसे देखा तो वे सोचने लगे कि यह चेलारम तो दूसरों की भांति सामान्य प्रतीत होता है। यह कैसे दूसरे लोगों के लिए मनहूसियत का कारण हो सकता है। इस बात को परखने के लिए उन्होंने आदेश दिया कि चेलाराम को उनके शयनकक्ष के सामने वाले कमरे में ठहराया जाए।	चेलाराम क्यों प्रसिद्ध थे?	hindi	अगर कोई सुबह-सवेरे उसका चेहरा सबसे पहले देख ले तो उसे दिनभर खाने को कुछ नहीं मिलता

After observing the results of our experiment, we can say RoBERTa fine-tuned on chaii data gives us an extractive output, that is it finds the output within the context only, thereby eliminating any scope of hallucination or false responses. This model can be implemented in settings where there is no room for error, like extracting facts from news articles, or key entities from an excerpt.

6. Conclusion and Future Expansion

Our work is a step towards improving the entire digital experience for Indian users and addressing the underrepresentation of Indian languages on the web. Through the utilization of the chaii-1 dataset and the RoBERTa model, we have exhibited the practicability and efficacy of precisely forecasting responses to inquiries in Hindi

As mentioned in the analysis, question and answering with RoBERTa is less prone to hallucination, and will be more viable in low computational environments where Large language models with retrieval augmented generation cannot be deployed

Furthermore, by presenting techniques and insights that can be applied to underrepresented languages globally, our experiment helps explore the field of multilingual natural language processing (NLP). We can promote more inclusivity and accessibility in digital communication by working together to create datasets and develop models. This will enable people with different language backgrounds to interact with online content more successfully and obtain information.

We anticipate more progress in multilingual NLU research in the future, with an emphasis on improving models and datasets specifically for Indian languages. Millions of people in India and around the world may benefit from new prospects for social, economic, and cultural empowerment if we keep bridging the language divide and supporting linguistic variety on the web.

7. References and Bibliography

[1] Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018).

[2] Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. "Roberta: A robustly optimized bert pretraining approach." *arXiv preprint arXiv:1907.11692* (2019).

[3] Wang, Luqi, Kaiwen Zheng, Liyin Qian, and Sheng Li. "A survey of extractive question answering." In *2022 International Conference on High Performance Big Data and Intelligent Systems (HDIS)*, pp. 147-153. IEEE, 2022.

[4] Kumar, Shailender. "BERT-based models' impact on machine reading comprehension in Hindi and Tamil." In *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, pp. 1458-1662. IEEE, 2022.

[] https://www.kaggle.com/competitions/chaii-hindi-and-tamil-question-answering