Group 9 : Mahabharata Information Retrieval

Akanksha Singh Arjun Singh Ayush Sahni Ayush Singh Neeraj Chouhan

21111005 21111402 21111019 21111020 21111044

{akankshas21, arjuns21, sayush21, ayushs21, neerajch21}@iitk.ac.in

Indian Institute of Technology Kanpur (IIT Kanpur)

April 2022

Abstract

Question answering systems is a wide and active area, with much scope of further research. In our project we developed an Information Retrieval system which is capable of providing answer to questions related to MAHABHARATA. We experimented on various setups of BERT and BM25. We also used pyserini LuceneSearcher to rank the documents, re-ranked it using BERT and then the topmost passage is used to retrieve answer using BertForQuestionAnswering.

1 Introduction

The Ramayana and the Mahabharata (Mahabharat in Hindi) can be considered as the greatest epics ever in the history of human civilization. They not only entertain the masses, but also enlighten them about duty morality and salvation.

These epics contain a lot of information about the ancient world and how people of that time used to live. Unfortunately, the two epics did not make it to the popular kid's literature, school curriculum, or popular folklore in the West. Even the western film world chose to ignore them completely because their themes and spiritual values are alien to the western world.

There should be easily and efficiently accessible sources of information from where people can learn these epics. But they are very huge to read. So, here we are presenting a Question-Answering system on Mahabharata, which for a given question retrieves its answer from the whole Mahabharata corpus.

For retrieving top-n passages, we used 'BM25', 'DistilRoberta' Bert, 'bert-base-nli-mean-tokens' and 'Pyrserini-LuceneSearcher' in different settings of our experiments.

For retrieving answers we used Question-Answering Bert model from The Hugging Face Library, which is already fine tuned on the SQuAD Dataset. We further Fine-tuned this model on our dataset.

2 Salient features of Dataset used

- Corpus: Complete English Mahabharata Corpus downloaded in text format.
  - The Corpus is divided into small chunks. Small chunks are divided in such that context does not break.
- We created Dataset by creating Question-Answer pair along with their context using English Mahabharta Corpus and Internet.

- In total we were able to create 270 Question-Answer Pair along with their contexts.
- This Question-Answer-Context dataset is split into train and test set for finetuning bert and for evaluation.

3 Methodology

3.1 DataSet Creation And Data Pre-processing

- Mahabharata corpus is divided into small chunks keeping in mind that context is not broken.
- Using question as a query, we have extracted top-n most relevant passages using Pyserini LuceneSercher, bert-base-nli-mean-tokens, nli-distilroberta-base-v2 Berts and Cosine Similarity. This Pyserini LuceneSercher takes input in JSON format. We already stored chunks(passages) in dictionary. Using a python script, we converted the dataset into required format.
- The topmost passage is sent to question-answer BERT as input to get the desired answer.

For Fine-tuning Question-Answering Bert :
Data needs to be in the specific format to fine-tune the Question-Answering Bert. The BERT is provided with the Question and Context along with 'start_answer' and 'end_answer' which are index from where the answer starts and ends in the context.

Since our Training Dataset is quite small, we fine-tuned our model only for 5 epochs, using 0.000001 as Learning rate and AdamW as optimizer to prevent over-fitting. Batch-size is set as 2 due to limited GPU resource.

3.2 Relevant Passage Retrieval System

3.2.1 Approach 1: Using BM25

- We stored the passages from the corpus into dictionary where key of dictionary corresponds to 'Doc_id' and value corresponds to passages.
- We passed the question as query to BM25 system, depending upon query, BM25 output top 10 relevant passages.
- These re-ranked passages are passed to Squad-Question-Answering Bert.

Insights : Since we know that BM25 operates on bag-of-words concepts which compares the terms in a query with the terms in a document. Hence its not necessary for the topmost document to be the relevant one. As a result, we did not get good results.

3.2.2 Approach 2: Using DistilRoberta based Encoder

- In this approach for finding question similarity with passages we used DistilRoberta-Bert Encoder instead of BM25.
- Here we pass all the documents to the DistilRoberta-Bert with returns 768 sized encodings(embeddings) for each Passage.
- Questions are also passed to DistilRoberta-Bert to find its encoding.
- Similarity between questions and passages is found using Cosine similarities of the encodings of questions and passages.
- Top-n Passages are returned based on the cosine-Similarity Scores.
- These Passages are sent to Squad-Question-Answering Bert, which returns the answer corresponding to each question.

Insights : This approach is based on the semantic meaning of sentences, as it uses BERT. This approach works comparatively better than Approach 1.

3.2.3 Approach 3: Using Bert-base-nli-mean-tokens based encoder

- This approach is pretty similar to the Approach 2, except that here we used Hugging face's sentence embedding model 'bert-base-nli-mean-tokens' for encoding passages.

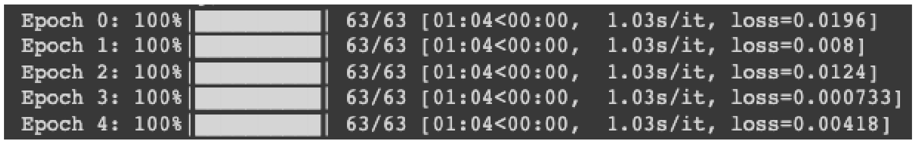
  Insights : Approach 2 and 3 are very similar, but the results that they provide are quite different. We observed that the encodings highlight different areas for encoding. The approach 2 gives better Relation-Based answers and approach 3 gives better answers for questions based on semantics.

3.2.4 Approach 4: Using Pyserini-Toolkit

- In our 4th approach, We used Pyserini toolkit from python for passage retrieval. Pyserini performs information retrieval research with sparse and dense representations.
- We used all three ways i.e. Sparse Retrieval, Dense Retrieval and Hybrid Retrieval to check which out of three performs best for our test dataset.
- Pyserini performs searching with the help of indexes which it creates for the input dataset. The input has to be in the JSON format, which we created with the help of a python script.
- Using the index file created by the encoders, the searcher performs the search to obtain 10 relevant passages.
- Pyserini also has Hybrid Encoder which exploits the feature of both sparse(retrieval using bag-of-words representations) and dense(retrieval using dense transformer-derived representations) which provides better performance.

3.3 Answer Retrieval System

- We have used The Hugging Face Transformers library which has a BertForQuestionAnswering model that is already fine-tuned on the SQuAD dataset. The Stanford Question Answering Dataset (SQuAD) is a collection of 100k crowdsourced QA pairs.
- The BertForQuestionAnswering class supports fine-tunning. This model is already fine tuned on SQuAD dataset, which gives pretty good results.
- We further Fine-Tuned the model using our training set. Since our Dataset is quite small, we Fine-Tuned the whole model only for 5 epochs and with low learning rate (0.00001) so that the model doesn't overfit on our model.



```
Epoch 0: 100%|██████████| 63/63 [01:04<00:00,  1.03s/it, loss=0.0196]
Epoch 1: 100%|██████████| 63/63 [01:04<00:00,  1.03s/it, loss=0.008]
Epoch 2: 100%|██████████| 63/63 [01:04<00:00,  1.03s/it, loss=0.0124]
Epoch 3: 100%|██████████| 63/63 [01:04<00:00,  1.03s/it, loss=0.000733]
Epoch 4: 100%|██████████| 63/63 [01:04<00:00,  1.03s/it, loss=0.00418]
```

Figure 1: Fine Tuning Bert

3.4 Working

BERT uses Transformer encoder blocks. The transformer encoder uses attention (Multi-Headed Self Attention) mechanism that learns contextual relations between words (or sub-words) in text.

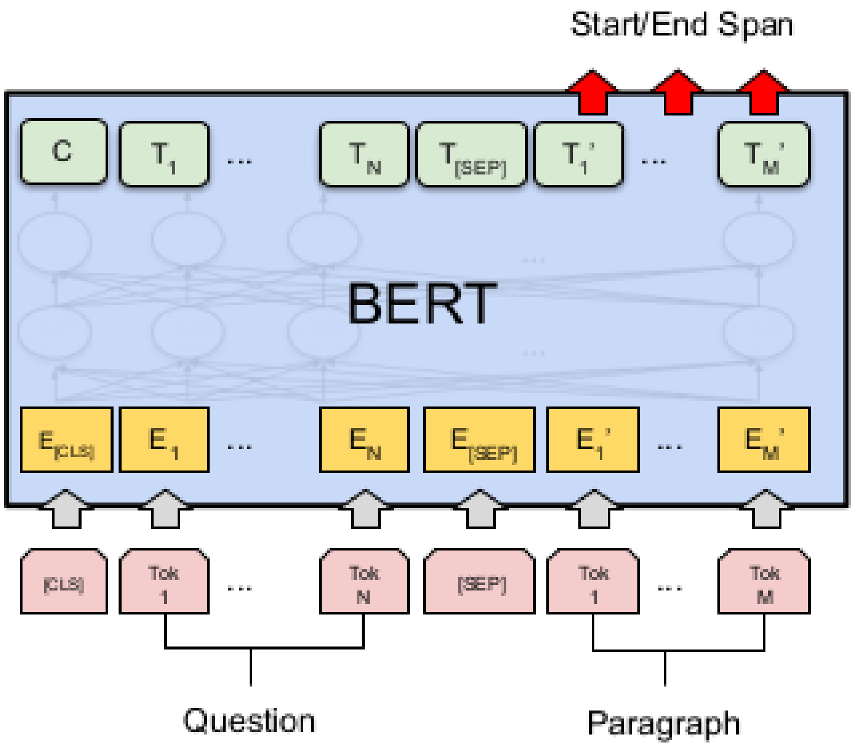


Figure 2: Inner working of Question Answering Bert model

For the Question Answering task, BERT takes the input question and passage as a single packed sequence. The input embeddings are the sum of the token embeddings and the segment embeddings. The input is processed in the following way before entering the model:

- **Token embeddings:** A [CLS] token is added to the input word tokens at the beginning of the question and a [SEP] token is inserted at the end of both the question and the paragraph.
- **Segment embeddings:** A marker indicating Sentence A or Sentence B is added to each token. This allows the model to distinguish between sentences. In the below example, all tokens marked as A belong to the question, and those marked as B belong to the paragraph.

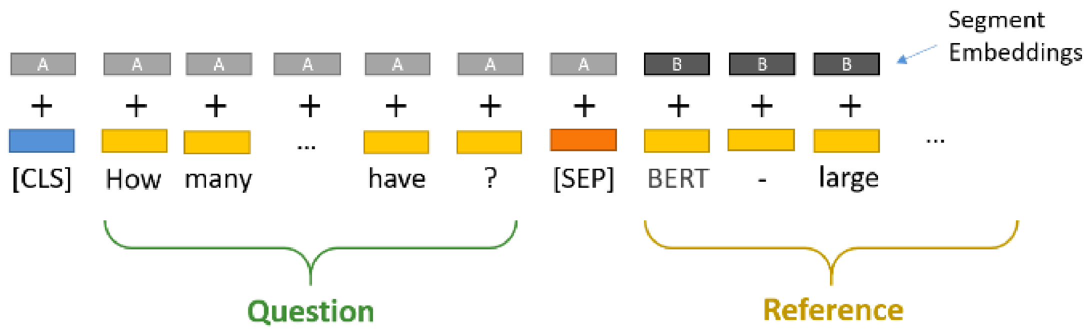


Figure 3:

To fine-tune BERT for a Question-Answering system, it introduces a start vector and an end vector. The probability of each word being the start-word is calculated by taking a dot product between the final embedding of the word and the start vector, followed by a softmax over all the words. The word with the highest probability value is considered. A similar process is followed to find the end-word.

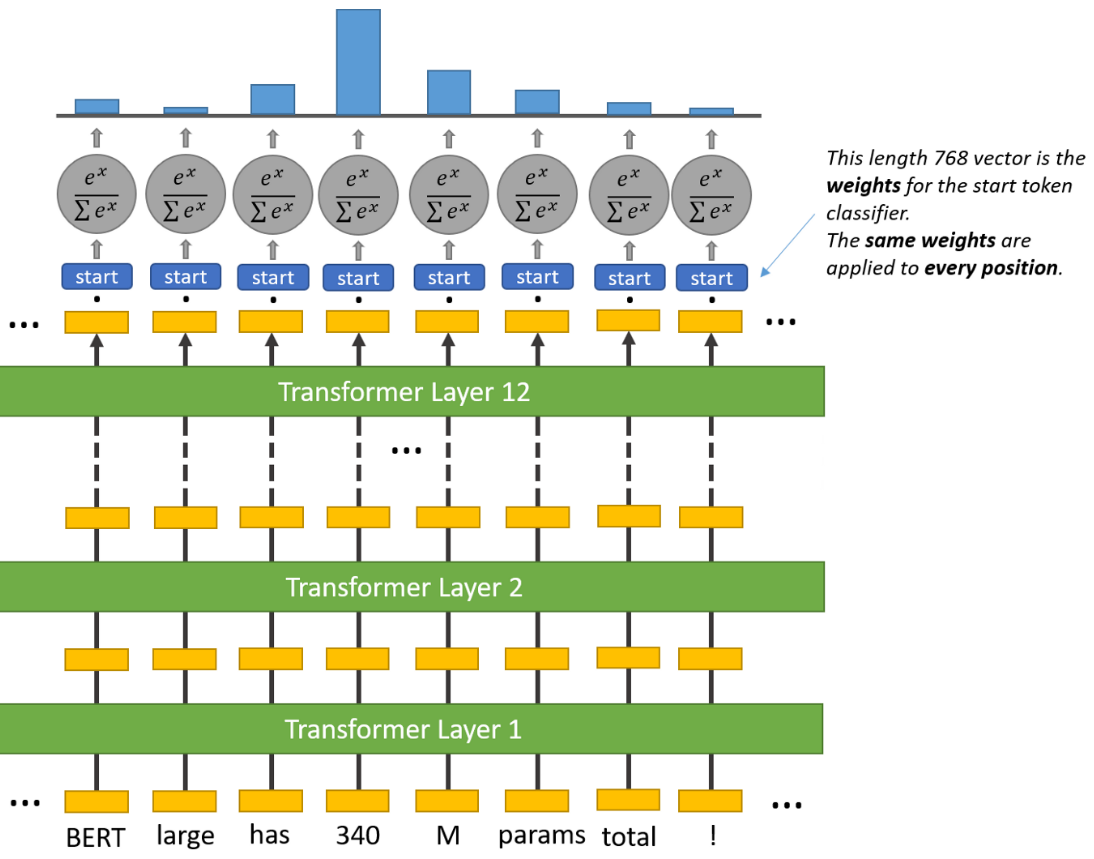

$\frac{e^x}{\sum e^x}$
start
This length 768 vector is the **weights** for the start token classifier. The **same weights** are applied to **every position**.
Transformer Layer 12
Transformer Layer 2
Transformer Layer 1
BERT
large
has
340
M
params
total
!

Figure 4:

3.5 Implementation Details

- We begin with collecting the dataset(MAHABHARATA.txt).We used this dataset in two ways:Passage creation and Manual Ques-Answer-Context Dataset Creation.
- we have used following Passage retrieval systems :
  - BM25
  - DistilRoberta BERT
  - bert-base-nli-mean-tokens BERT
  - LuceneSearcher
- We were generating 10 passages ranked according to the score they obtained using cosine similarities (in case of Approach 2,3,4) and BM25 Score (in case of approach 1).
- Passages retrieve passes top 2 passages to BertForQuestionAnswer as it can take 512 tokens at a time ,hence passages were chunked accordingly.Two passages at a time was send to improve the result.So if two passages together creates 512 tokens or lesser, they together as single string was passed to the model to predict the output.

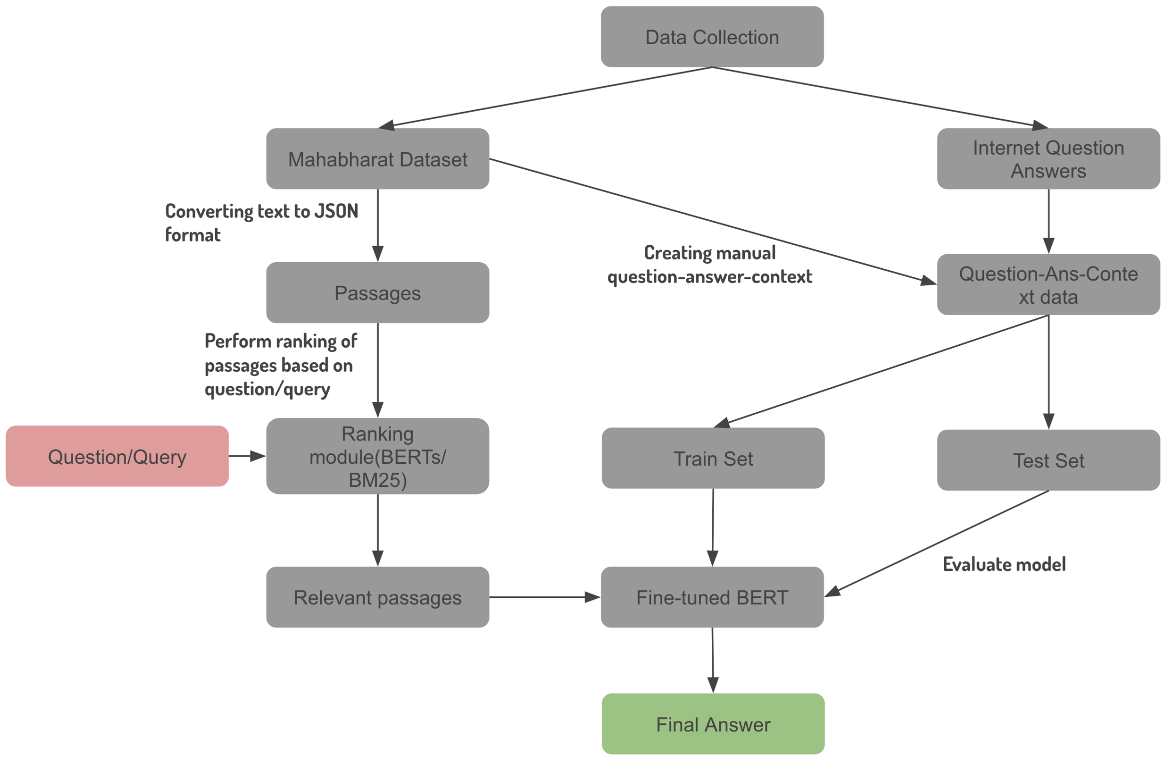

Data Collection
Mahabharat Dataset
Internet Question Answers
Converting text to JSON format
Creating manual question-answer-context
Passages
Question-Ans-Context data
Perform ranking of passages based on question/query
Question/Query
Ranking module(BERTs/ BM25)
Train Set
Test Set
Evaluate model
Relevant passages
Fine-tuned BERT
Final Answer

Figure 5: The flow diagram attached gives the clear understanding of the system's working.

4 Experimental Results

We used test dataset to check the accuracy of our IR System. As we can see, Approach-4 performed pretty well resulting 79% top-1 Accuracy.

Approach	Top-1-Accuracy	Top-5-Accuracy
Approach-1 {BM25 based}	15%	33%
Approach-2 {DistilRoberta based Encoder}	31%	68%
Approach-3 {Bert-base-nli-mean-tokens based encoder}	25%	43%
Approach-4 {Pyserini-Toolkit based}	**79%**	**84%**

Some test questions are:

- Who was the daughter of king subala?
- What What benediction by lord shiva gandhari was granted?
- Pritha was adopted by whom?
- Whom did kunti marry to?
- Dhritarastra was married to whom?

4.1 Success Cases

- Q1-Who was the daughter of king subala?
  Ans-gandhari
- Q2-What benediction by lord shiva gandhari was granted?
  Ans-that she could have one hundred sons
- Q3-Pritha was adopted by whom?
  Ans-king kuntibhoja

4.2 Failure Cases

- Q1-Whom did kunti marry to?
  Ans-karna
- Q1-Dhritarastra was married to whom?
  Ans-pandu

We are getting some failure cases mainly because some passages are contextually similar hence the score generated by the searcher is very close for them,hence Our system is not so accurate in such scenarios.

5 Future Work

We successfully created a Question-Answering system using BERT but there are still few areas which can be improved.

- We can incorporate various other Epics like Ramayana, Bhagwat Geeta, etc and create a single model to Answer Question based on different Epics.
- Currently our model reads English Mahabharta corpus and returns answers in English Language. Since Mahabharta was originally written in Sanskrit Language, and better translations are available in Hindi Language, work can be done to transfer Model to answer the questions in Hindi/Sanskrit Language using Hindi/Sanskrit Corpus.
- Currently Our model takes 3-4 secs to answer a single question, which is little slow of real time question answering. A faster and more efficient approach can be used to make our model faster.

6 Conclusion

We successfully developed a Question-Answering system for Mahabharata. We can conclude that question answering system depends heavily on retrieving relevant documents. We Observed that Bert based approaches for retrieving the passage works a lot better than Bag-of-word based techniques like BM25 which shows that semantic meaning is very important for Question-Answering Systems, such as ours. Since different Bert Models uses different techniques for encoding a passage. It results in different similarity score for the same question-passage pair. These different scores results a particular model performing better for some set of questions.

We can conclude that the Encoding Bert models can be fine tuned on specific dataset for better results. Our best Model(Approach 4) obtained 79% Top-1-accuracy on the dataset.

7 References

- Pyserini- An Easy-to-Use Python Toolkit to Support Replicable IR Research with Sparse and Dense Representations
  https://arxiv.org/abs/2102.10073
- Faiss: A library for efficient similarity search
  https://engineering.fb.com/2017/03/29/data-infrastructure/faiss-a-library-for-efficient-similarity-search/
- pyserini 0.16.0
  https://pypi.org/project/pyserini/
- Huggingface-Transformers
  https://huggingface.co/docs/transformers/index
- Question Answer set
  - https://www.funtrivia.com/trivia-quiz/Religion/The-Great-Mahabharatha-366607.html
  - https://www.funtrivia.com/trivia-quiz/Religion/Mahabharata-347519.html
  - https://www.funtrivia.com/trivia-quiz/Religion/Mahabharata—Kingship-Knows-No-Kinship-328402.html
  - https://www.funtrivia.com/trivia-quiz/Religion/The-Mahabharata-283372.html
  - https://www.funtrivia.com/trivia-quiz/Religion/The-Great-War-The-Mahabharatha-II-150304.html
  - https://www.funtrivia.com/trivia-quiz/Religion/Mahabharata–The-Greatest-Epic-Part-1-102714.html
  - https://www.funtrivia.com/trivia-quiz/Religion/Mahabharata–The-Greatest-Epic-Part-2-103090.html
  - https://www.funtrivia.com/trivia-quiz/Religion/Mahabharata–The-Great-Indian-Epic-223512.html
- Blog on BERTs
  https://jalammar.github.io/illustrated-bert/
- BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
  https://arxiv.org/pdf/1810.04805.pdf
- CoQA: A Conversational Question Answering Challenge
  https://arxiv.org/pdf/1808.07042.pdf
- Distilbert
  https://huggingface.co/distilbert-base-uncased
- Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks
  https://arxiv.org/abs/1908.10084