IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 7, July 2022

DOI: 10.17148/IJARCCE.2022.11795

Natural Language Processing-based Reading Comprehension and Question Answering Model

Harshit Handa[1], Kushagra Gupta[2] and Merin Meleet[3]

[1,2,3] R.V. College of Engineering, Bengaluru, Karnataka, 560059

Abstract Numerous machine learning techniques aim to respond to queries in natural language.Back then, people frequently used bags of words to attempt to address questions something the developers had already predetermined. Using this approach, programmers must invest a lot of the questions and answers for the specific questions that took time to write. This approach was incredibly helpful but was unable to respond to queries for the massive database for the chat bots. Current Natural Transformer-based models predominate in language processors. Making use of these transformers HuggingFace introduced the Bert Quality Assurance approach, which reads through the text, in libraries the user-provided textual context and tries to respond to queries pertaining to that textual context.This technique has shown promise in addressing the intricate query from a lengthy document. Users can simply ask inquiries about a specific year or the profits they made for that year, for instance, if a corporation provides a report regarding their financial years that has been passed through the model. And the solution can be found in a matter of seconds without scrolling through the documents.

Keywords: BERT, Question-Answering model, sequence-to-sequence model

1 INTRODUCTION

This technique has shown promise in addressing the intricate query from a lengthy document. Users can simply ask inquiries about a specific year or the profits they made for that year, for instance, if a corporation provides a report regarding their financial years that has been passed through the model. And the solution can be found in a matter of seconds without scrolling through the documents.They are able to respond in our everyday normal language. In addition to providing responses, it makes an effort to accurately comprehend the questions' intentions and searches for the specific phrase or sentence it needs. To do this, I'll be using the "Bert Model from Hugging Face Transformers," which can be downloaded and used from the Hugging Face webpages. To accomplish this, I'll also be using a variety of Python libraries.Machine (Reading) In the NLP area of comprehension, we train computers to comprehend and respond to queries using unstructured text. Over 100,000 question-answer pairs derived from Wikipedia articles made up the SQuAD (Stanford Question Answering Dataset) dataset, which was created by StanfordNLP in 2016. It was difficult to teach a machine learning model to respond to queries using information from a contextual document. The model would return the subset of the text most likely to answer the query when given a contextual document (free form text) and a question.

2 LITERATURE REVIEW

Jacob Devlin and team in [1] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.Bidirectional Encoder Representations from Transformers, or BERT, is a brand-new language representation paradigm that we offer. By concurrently conditioning on both left and right context in all layers, BERT is aimed to pre-train deep bidirectional representations from unlabeled text, in contrast to recent language representation models.

Karl Moritz Hermann and team in [2] Teaching Machines to Read and Comprehend. Developed a new technology that addresses this bottleneck and offers supervised reading comprehension data on a big scale. As a result, a subset of attention-based deep neural networks that require little to no prior understanding of linguistic structure were able to learn to read real documents and respond to challenging questions.

Jason Weston and team in [3] Towards AI-Complete Question Answering : A Set Of Prerequisite Toy Tasks. proposed a framework and a collection of artificial tasks (the bAbI tasks) to aid in the creation of learning algorithms for text comprehension and reasoning.

DOI: 10.17148/IJARCCE.2022.11795

Jason Weston and team in [4]Memory Networks. The model works for an image having a single person as well as multiple persons. Memory networks employ long-term memory and inference components to reason; they can combine these two functions. It is possible to read from and write to the long-term memory with the intention of using it for prediction.In the context of question-answering (QA), where the long-term memory effectively serves as a (dynamic) knowledge base and the output is a textual response, authors looked into various models.

Sainbayar Sukhbaatar and team in [5] End-To-End Memory Networks . The authors of this research have introduced a revolutionary recurrent neural network (RNN) architecture in which the recurrence repeatedly reads from a potentially huge external memory before producing a symbol. We can think of our model as a continuous version of the Memory Network.

Ankit Kumar. and team in [6] Ask Me Anything: Dynamic Memory Networks for Natural Language Processing.. The dynamic memory network (DMN), a neural network design that analyses input sequences and queries, creates episodic memories, and generates pertinent answers, has been introduced in this research. An iterative attention process is triggered by questions, allowing the model to focus its attention on the inputs and outcomes of prior iterations.

Venkatesh Krishnamoorthya in [7] Evolution of Reading Comprehension and Question Answering Systems. This essay charts the development of reading comprehension and question-answering techniques. Traditional NLP methods were used first, followed by contextualized word/sentence embeddings with ELMo. After looking at BERT, Attention, Transformer, and Open Domain Question Answering with HayStack Neural Search, we moved on to more recent research on Multi-Hop QA and Open Domain Question Answering.

Salima Lamsiyah and team in [8]. Unsupervised extractive multi-document summarization method based on transfer learning from BERT multi-task fine-tuning. The BERT model is fine tuned on intermediate tasks from the GLUE benchmark using single-task and multi-task fine tuning; after examining the impact of both methods, BERT multi-task fine-tuning achieves a significant performance improvement. This paper proposed an unsupervised method for extractive multi-document summarization based on transfer learning.

3 METHODOLOGY

We implemented the baseline model: the standard BERT model applied on the SQuAD dataset as described in the paper by Devlin et al. [2018]. The strategy is to improve the BERT model by adding a second linear layer to forecast the beginning and end of the response. For the baseline model, we reused the existing PyTorch version of BERT made by NLP researchers from HuggingFace. The code script provided from HuggingFace is a giant file with 1085 lines, which made it very difficult to make improvements. First we refactored the code into multiple modules. Then we added checkpoint saver and loader with scores evaluated on the val set. Next we implemented our extensions as described next. We improved the baseline model with plausible answers as input and proposed a Question-Answer Evaluator as inspired by the Read+Verify paper. The reader will predict plausible answer for even unanswerable questions and the evaluator will check whether the candidate answer is legitimate.

- Input data: To improve the training of the other parts of our technique, we adjusted the data processing script to include not only questions with answers but also cases with no answers as examples. We also considered the plausible answer to be the best answer.
- Comprehension Reader: Using the question and paragraph as input, we first converted them into a series of tokens (question and response tokens), which we then sent into the reader's embedding layer to create token embeddings using the BERT model. Next, using the same strategy as in the baseline model, the linear layer of the reader reads the generated hidden vector and outputs two features: start and end positions. The cross entropy loss for the start and finish points serves as the loss function. Adam is the optimizer throughout training, and the loss is averaged over the batch.
- Question-Answer Evaluator: Given a question, a paragraph, and answers as input, the evaluator will first identify the sentence in the paragraph that contains the answer. The embedding layer will then process this information into a sequence of tokens (question and answer sentence tokens), and create an embedding for each token using the BERT model. The pooled vector generated by BERT is read by the classification layer of the evaluator, which then produces a no-answer probability at the sentence level. To reduce the negative log-likelihood of the right labels, it uses a typical cross-entropy objective. We developed the evaluator model to use BERT to forecast the probability of a no-answer, constructed the data processing code to turn data into training examples, and created the script to carry out training, evaluation, and testing.

● In order to forecast a response, we first broke down the input into a series of tokens (questions and answers), which we then fed into the reader model to provide a candidate answer and a no-answer probability at the passage level. The evaluator model calculates a sentence-level probability by finding the answer sentence given the candidate response, processing it with the question to create a token sequence (question and answer sentence tokens), and feeding it to the evaluator model.

4 RESULTS AND ANALYSIS

Test process

The act of testing involves running a software to look for flaws. Our software must be error-free in order to function properly. The software will be free of all errors if testing is successful.

● The smallest component of software design is the subject of unit testing. Here, we test a single unit or a collection of related units. The programmer frequently does this task by examining the results of sample inputs and outputs.

● The goal of integration testing is to use unit-tested parts to assemble a programme structure that has been determined by design. Integration testing involves combining a number of components to produce an output.

● Regression testing: Whenever a new module is added, the programme is altered. This kind of testing ensures that the entire component functions correctly even after other components are added to the entire application.

● Alpha testing: This kind of validation testing is known as alpha testing. It is a kind of acceptability testing that is carried out prior to the product being made available to customers. QA personnel often carry it out.

● Beta testing: The software's end-user does the beta test at one or more client locations. This version is made available for a small group of users to test in a live setting.

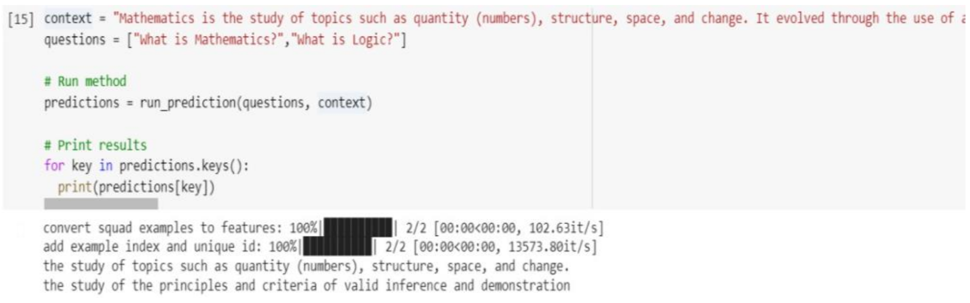
● System testing: This software has been thoroughly tested to ensure that it runs well on all platforms. It is included in the black box testing methodology. In this, we only pay attention to the necessary input and output, ignoring internal functioning.

● Stress testing is putting the system under challenging circumstances to see how it responds.

● Acceptance testing: Customers do acceptance testing to see whether the provided products fulfil the requirements or not.

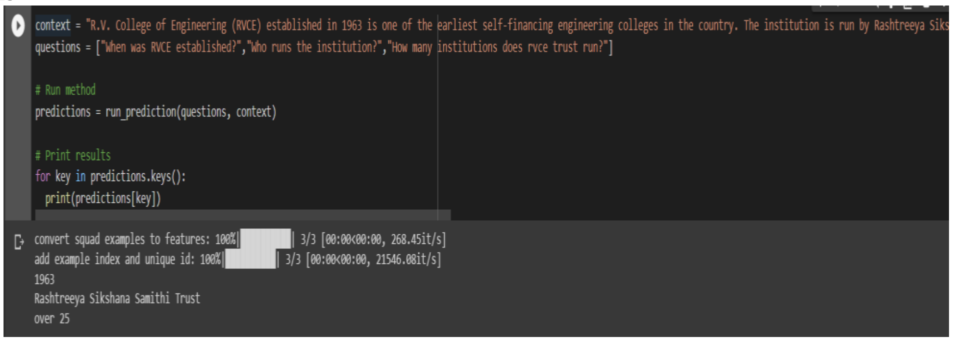
Test cases
Test case 1:

Basic Questions



```
[15] context = "Mathematics is the study of topics such as quantity (numbers), structure, space, and change. It evolved through the use of a
     questions = ["What is Mathematics?","What is Logic?"]

     # Run method
     predictions = run_prediction(questions, context)

     # Print results
     for key in predictions.keys():
       print(predictions[key])

     convert squad examples to features: 100%|██████████| 2/2 [00:00<00:00, 102.63it/s]
     add example index and unique id: 100%|██████████| 2/2 [00:00<00:00, 13573.80it/s]
     the study of topics such as quantity (numbers), structure, space, and change.
     the study of the principles and criteria of valid inference and demonstration
```

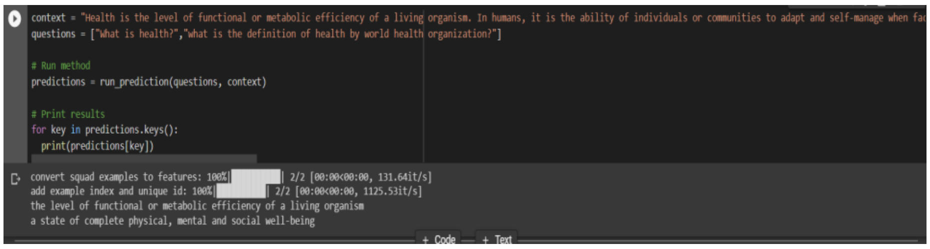
Test Case 2:
Basic Questions



```
context = "R.V. College of Engineering (RVCE) established in 1963 is one of the earliest self-financing engineering colleges in the country. The institution is run by Rashtreeya Siks
questions = ["When was RVCE established?","Who runs the institution?","How many institutions does rvce trust run?"]

# Run method
predictions = run_prediction(questions, context)

# Print results
for key in predictions.keys():
  print(predictions[key])

convert squad examples to features: 100%|██████████| 3/3 [00:00<00:00, 268.45it/s]
add example index and unique id: 100%|██████████| 3/3 [00:00<00:00, 21546.08it/s]
1963
Rashtreeya Sikshana Samithi Trust
over 25
```

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⵂ Impact Factor 7.39 ⵂ Vol. 11, Issue 7, July 2022
DOI: 10.17148/IJARCCE.2022.11795

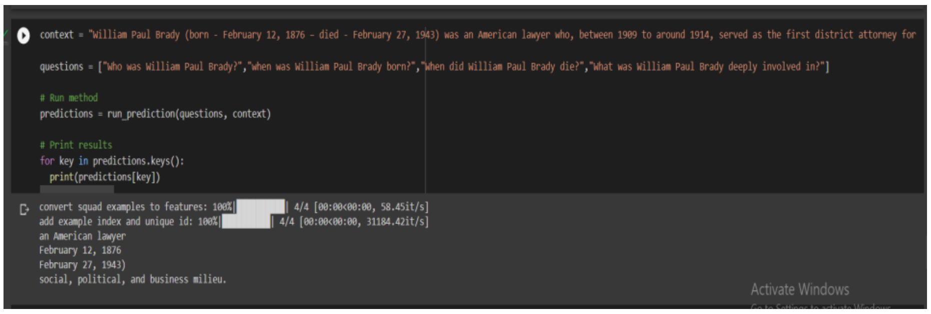
Test Case 3:
Intermediate Questions



```
context = "Health is the level of functional or metabolic efficiency of a living organism. In humans, it is the ability of individuals or communities to adapt and self-manage when fac
questions = ["What is health?","what is the definition of health by world health organization?"]

# Run method
predictions = run_prediction(questions, context)

# Print results
for key in predictions.keys():
  print(predictions[key])
```

```
convert squad examples to features: 100%|██████████| 2/2 [00:00<00:00, 131.64it/s]
add example index and unique id: 100%|██████████| 2/2 [00:00<00:00, 1125.53it/s]
the level of functional or metabolic efficiency of a living organism
a state of complete physical, mental and social well-being
```

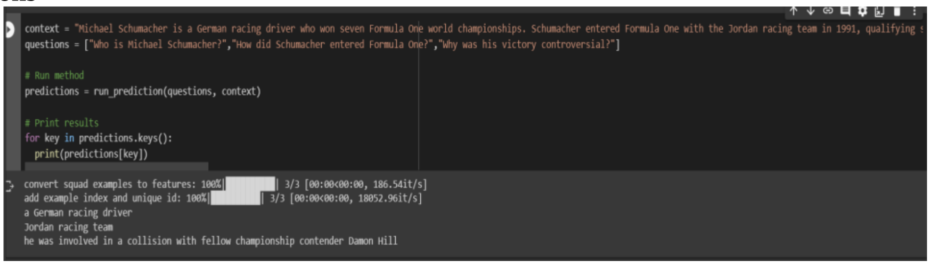
Test Case 4:
Tough Questions



```
context = "William Paul Brady (born - February 12, 1876 - died - February 27, 1943) was an American lawyer who, between 1909 to around 1914, served as the first district attorney for

questions = ["Who was William Paul Brady?","when was William Paul Brady born?","When did William Paul Brady die?","What was William Paul Brady deeply involved in?"]

# Run method
predictions = run_prediction(questions, context)

# Print results
for key in predictions.keys():
  print(predictions[key])
```

```
convert squad examples to features: 100%|██████████| 4/4 [00:00<00:00, 58.45it/s]
add example index and unique id: 100%|██████████| 4/4 [00:00<00:00, 31184.42it/s]
an American lawyer
February 12, 1876
February 27, 1943)
social, political, and business milieu.
```

Test Case 5:
Tougher questions

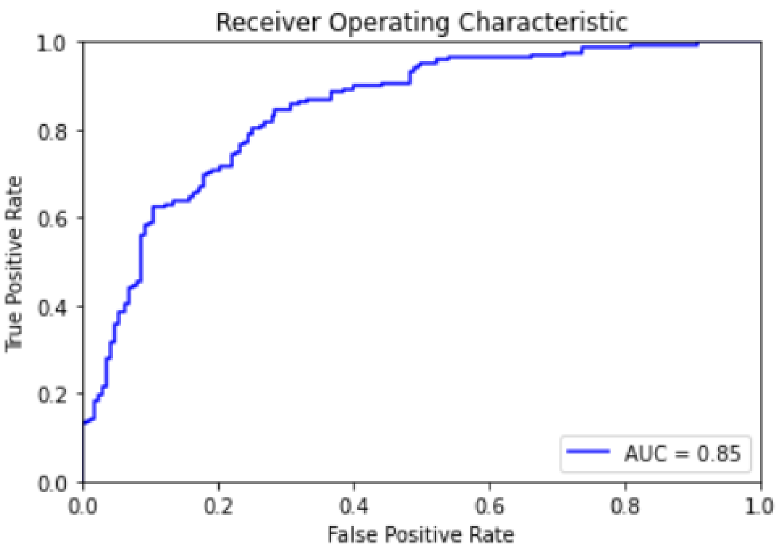


```
context = "Michael Schumacher is a German racing driver who won seven Formula One world championships. Schumacher entered Formula One with the Jordan racing team in 1991, qualifying s
questions = ["Who is Michael Schumacher?","How did Schumacher entered Formula One?","Why was his victory controversial?"]

# Run method
predictions = run_prediction(questions, context)

# Print results
for key in predictions.keys():
  print(predictions[key])
```

```
convert squad examples to features: 100%|██████████| 3/3 [00:00<00:00, 186.54it/s]
add example index and unique id: 100%|██████████| 3/3 [00:00<00:00, 18052.96it/s]
a German racing driver
Jordan racing team
he was involved in a collision with fellow championship contender Damon Hill
```

RESULTS

The trained Bert model achieved a combined accuracy of 75.59% and AUC as 0.85.



DOI: 10.17148/IJARCCE.2022.11795

5 CONCLUSION AND FUTURE WORK

BERT is capable of comprehending linguistic structure and managing interdependencies between phrases in addition to the duty of answering questions. It may respond to the queries using straightforward logic. Which college does John's sister attend? can be answered using BERT, as was described in the aforementioned case. appropriately in the sentence "John is a 10-year-old boy" without mentioning John's sibling. He is Robert Smith's son. Robert's spouse is Elizabeth Davis. She is UC Berkeley's professor. Elizabeth Smith's daughter is Sophia Smith. She goes to UC Davis to study. Also keep in mind that the longer passage can be run through the model, but the code will automatically truncate the extra part if the length of the question and passage exceeds 512 tokens.There is still room to enhance this method with more datasets and more potent BERT models. For even greater accessibility, the programme might be created as a web-based application. These models can be utilized in a variety of settings, including querying lengthy corporate data and chatbots.

REFERENCES

1. Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
2. Hermann, Karl Moritz, et al. "Teaching machines to read and comprehend." Advances in neural information processing systems 28 (2015).
3. Weston, Jason, et al. "Towards ai-complete question answering: A set of prerequisite toy tasks." arXiv preprint arXiv:1502.05698 (2015).
4. Weston, Jason, Sumit Chopra, and Antoine Bordes. "Memory networks." arXiv preprint arXiv:1410.3916 (2014).
5. Sukhbaatar, Sainbayar, Jason Weston, and Rob Fergus. "End-to-end memory networks." Advances in neural information processing systems 28 (2015).
6. Kumar, Ankit, et al. "Ask me anything: Dynamic memory networks for natural language processing." International conference on machine learning. PMLR, 2016.
7. Krishnamoorthy, Venkatesh. "Evolution of Reading Comprehension and Question Answering Systems." Procedia Computer Science 185 (2021): 231-238.
8. Lamsiyah, Salima, et al. "Unsupervised extractive multi-document summarization method based on transfer learning from BERT multi-task fine-tuning." Journal of Information Science (2021): 0165551521990616.