SMU Data Science Review

Volume 3
Number 3 *Fall 2020*

Article 3

2020

BERT for Question Answering on BioASQ

Eric R. Fu
Southern Methodist University, efu@smu.edu

Rikel Djoko
Southern Methodist University, rdjoko@mail.smu.edu

Maysam Mansor
Southern Methodist University, mansor@mail.smu.edu

Robert Slater
SMU, rslater@mail.smu.edu

BERT for Question Answering on BioASQ

Eric Ross Fu[1], Maysam Mansor[1], Rikel Djoko[1], Robert Slater[2]
[1] Master of Science in Data Science, Southern Methodist University,
Dallas, TX 75275 USA
[2] Department of Data Science Southern Methodist University
Dallas, TX 123456 USA
{efu, mansor, rdjoko, rslater}@smu.edu

Abstract. Machine reading comprehension and question answering are topics of considerable focus in the field of Natural Language Processing (NLP). In recent years, language models like Bidirectional Encoder Representations from Transformers (BERT) [3] have been very successful in language related tasks like question answering. The difficulty of the question answering task lies in developing accurate representations of language and being able to produce answers for questions. In this study, the focus is to investigate how to train and fine tune a BERT model to improve its performance on BioASQ, a challenge on large scale biomedical question answering. Our most accurate BERT model achieved an F1 score of 76.44 on BioASQ, indicating successful performance in biomedical question answering.

1 Introduction

Question Answering (QA) systems are becoming increasingly necessary and popular because many problem domains can benefit from having their questions answered in a convenient and time effective manner. Nowadays society is encountering larger volumes of information and also questions. It can be a challenge for organizations to answer large volumes of questions because getting answers from employees requires an expenditure of time and expertise. This is a burden for the asker and the answerer. QA systems can offer the organizations the ability to accurately answer large volumes of questions while minimizing the cost.

In many scenarios, organizations have information, but struggle to locate and distribute answers. Society faces huge volumes of incoming information and it can be a challenge to keep up in understanding this data. QA systems are a convenient way of finding and providing answers among huge volumes of textual information. QA systems utilize large volumes of text data to train its language model to answer questions related to the text. NLP methods give our model an understanding of the text and allow our model to process questions through its own understanding, which will result in an answer [2].

There are several types of question answering systems. For some questions, a search on the internet is all that is necessary. Yet, this type of system can only make use of generally available information. Open domain QA systems parse the internet for results

related to a question. Alternatively, they utilize a database of information to draw from. In closed domain QA, a question of interest is not generally available for the internet to answer. Closed domain QA systems must be trained on unique documents in order to provide question answering related to those documents.

Although the BioASQ dataset is publicly available it is considered a closed domain problem. This study will illustrate how BERT could be applied to a closed domain QA scenario. This research will use NLP and deep learning techniques to train a BERT QA model for language comprehension on biomedical data from BioASQ. The proposed model system will be based on a BERT architecture that was pre-trained using BioASQ and also SQUAD dataset, which is another popular QA challenge dataset. This study will use a transfer learning technique to apply our BERT model to the test questions. The focus of this research paper is on applying the details of the BERT model to create an accurate QA system within a closed domain of knowledge.

2 Literature Review

For the past two decades QA has been a fast growing area of research in the field of Computer Science and Natural language processing (NLP).

QA systems must receive text, understand text questions, search for passages within text, and output text answers. So, in this field there is a need for NLP techniques. NLP is a field of Machine learning that uses computers to process and analyze human language. Building an NLP pipeline requires data cleaning, data pre-processing, word and sentence representation in vector form and fitting it to the machine learning model for prediction.

The challenge of using computers to process language is that computers aren't designed to understand word meaning. In order for a computer to understand text data, it is standard for features of the text to be represented as numbers.

Word meanings in computers are able to be represented with Glove Vectors [16], a pretrained dataset of word vectors that are individually meant to represent word meanings. This data was developed by training for word context over Wikipedia datasets and is frequently used in NLP to help models represent word meaning.

Document similarity can be represented with Tf-Idf, which refers to a calculation of term frequency-inverse document frequency. The term frequency is the raw count of a term in the document, and the inverse document frequency is the ratio or probability of a particular word in the document.

Similar works have incorporated a three stage architecture for building question answering systems [10]. The first phase consists of question processing which consists of taking a user question and breaking it down in a way that the machine can understand the underlying meaning. The second stage is document retrieval, which uses the translated question to retrieve relevant documents. The last stage is the answer processing, which will use the selected documents to pick the best answer. Beside the common high-level architecture, they all have different implementations based on the paradigm of the researcher. There are three major implementation paradigms of

question answering: information retrieval QA, knowledge base QA and the Deep learning QA.

The information retrieval QA, given a question, uses information retrieval techniques to extract passages directly from these documents [12]. Knowledge base QA is bridging the gap between natural language expressions and the complex schema of the knowledge base. Build a semantic representation of the query and use this representation to query databases of facts [4]. It's an effective method which decomposes the user's natural language questions and extracts the keywords and conditions automatically [5].

In deep learning QA, this is the most recent approach which uses NLP and machine learning to outperform the retrieval-based and knowledge based on more challenging questions[6]. Within deep learning there has been different technique such as: recurrent neural networks (RNN), gated recurrent neural networks (GRU) [18], long short term memory networks (LSTM) [17], and most recently Transformer models.

Deep learning is a subset of machine learning that deals with stacking and combining the powers of multiple predictors. A neural network is a connection of layers which consist of 1 or more layers of predictors. This concept allows neurons to take an input X , process it using an activation function and return a value output, which is again processed through another layer to produce a final result.

The RNN is a specialized deep learning approach that remains a popular method at processing sequential information like language and audio data. RNNs are popular for this task because they process inputs in a sequential manner and are simple to tune and train compared to other deep learning methods.

Sequential model algorithms have made exciting progress in the last years, enabling numerous exciting applications such as speech recognition, question answering, chatbots and language translation. Variations of the RNN include the GRU and LSTM, which have also been used extensively in language tasks like QA.

The attention mechanism is one of the most recent approaches in deep learning which is based on the encoder-decoder architecture with a multi-layer and multi-headed -attention [15].

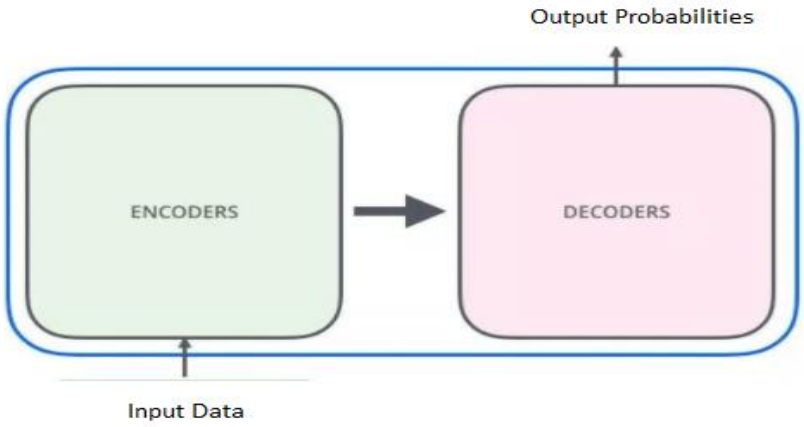

Fig 1. Demonstrating a simplified overview of the transformer architecture's encoder to decoder flow.

This approach outperforms the traditional RNN model because the sequential processing nature of RNNs is time consuming and is resistant to the advantages of parallel computing. The attention mechanism calculates a context vector that captures additional information from the inputs. For example, in a text processing scenario, attention will capture an input word's relation to every other word in the input. Compared to RNNs, this method boasts a higher potential in language comprehension tasks. This is because the attention mechanism captures language context more effectively than RNN. Attention mechanisms can differ in the subject of the context calculated and what is calculated based on the context.

Transformers are currently state of the art in NLP tasks, outperforming RNNs. Transformer models are based solely on these attention mechanisms. Typically, these models belong to a family of encoder decoder systems.

Transformer models work by stacking attention mechanisms on top of each other. A transformer's encoder takes an input text and calculates encoder embeddings. The encoder can use self-attention by considering the context from itself while making calculations for the encoder embedding. The context from these encoder embeddings are inputted into the decoder through a separate attention mechanism. Transformer models have several separate attention mechanisms for the sake of capturing as much context possible from language data.

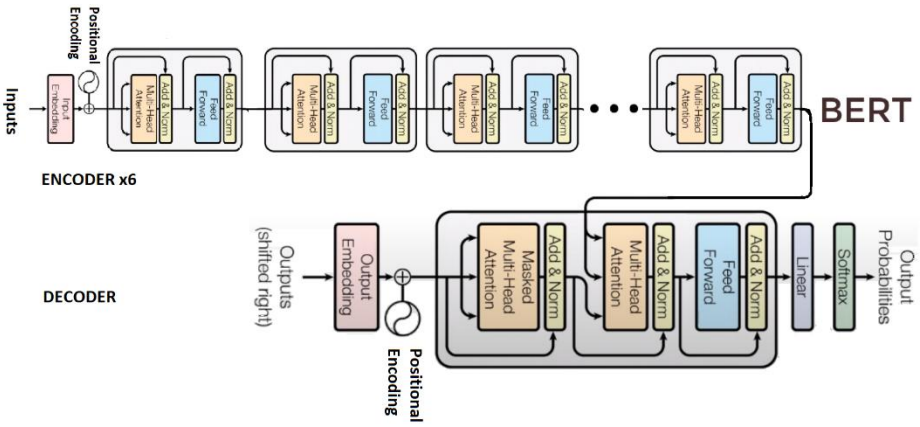

Fig 2. The BERT architecture involves multiple encoders which use attention mechanism [15].

BERT is a conceptually simple but empirically powerful transformer model based on attention mechanisms. Its performance is state of the art in a wide range of NLP tasks [7].

BERT is a Bi-Directional Attention Flow (BIDAF) network, a multi-stage hierarchical process that represents the context at different levels of granularity and uses

bidirectional attention flow mechanism to obtain a query-aware context representation without early summarization [7].

End-To-End memory networks, which is a recurrent neural network with attention model over a large external memory, where the model is reading from [14]. This may involve fitting an LSTM on top of a BERT model for additional performance increase. This approach is doing a good job at QA question since it's considering the dependency and the out of order within the dataset. Notably, this method adapts well when impossible questions are involved.

Yet, there's no QA systems designed to address a variety of domains such as customer support questions for software products. The existing works helped in understanding and developing the solution. This study aims to fill the gap on open domain QA and close domain QA using BERT as it is the highest performing technique available.

3.0 Methodology

3.1. Data

In this research, the primary objective is to accurately answer questions related to the BioASQ dataset. BioASQ is an EU-funded biomedical semantic indexing and question answering challenge that provides accumulated sets of biomedical questions and gold standard answer data. Questions within the BioASQ data are associated with scientific articles from PubMed and GoPubMed, which are journals for publishing scientific research. Consequently, the BioASQ dataset contains large volumes of biomedical scientific articles that must be semantically indexed for the purpose of training an NLP model to answer questions related to the text. There are four types of questions in the BioASQ: factoid questions, summary questions, yes/no questions, and summary questions.

Notably, our study also used the Stanford Question Answering Dataset (SQuAD) to train the QA model. SQuAD is a reading comprehension dataset, consisting of questions posed by crowd workers on a set of Wikipedia articles, where the answer to every question is a segment of text, or *span*, from the corresponding reading passage, or the question might be unanswerable. SQuAD 2.0 combined the 100,000 questions in SQuAD 1.1 with over 50,000 unanswerable questions written in adversarial fashion by crowd workers to look similar to answerable ones. To do well on SQuAD 2.0, systems must not only answer questions when possible, but also determine when no answer is supported by the paragraph and abstain from answering.

The limitation of SQuAD dataset is that is not performing well on closed domain. It mainly uses general knowledge generated from a set of Wikipedia articles. This paper addressed that limitation by defining a customized question answering systems for Biomedical question using the BioASQ dataset.

The BioASQ dataset can be obtaining by registering online at the BioASQ website which authorizes the use of the dataset. The dataset can then be unpacked to json format. The dataset contains an array of questions where each question is represented as a json object where each question is represented by a body which contain the actual question. Each question is associated with several other features. One of which is the ideal answer for the question. This is the label for each data point so that the QA model is a

supervised machine learning model. Also, several documents are associated with each question. The documents are scientific research papers which are related to the topic of the question. The abstract text from these documents serves as our model's context when trying to predict an answer to each question. In order to correctly answer a question, the model must be given a passage of text that contains the necessary information to answer the question. That is why the dataset has another feature named 'is_impossible'. This feature helps the model's overall language comprehension, by potentially helping it understand why a correct answer was not found.

The association of question, context, and answer within the dataset is a useful format for training QA. The specifically scientific topics in the dataset help train the QA model for that the specific task of answering scientific questions. The dataset overall is highly specialized for our specific task. There are 11,000 question entries and several different types of questions. Some of each of these question types cannot be answered based on the reference text. Here in Table 1, the diversity of question types in the BioASQ dataset and their frequency is demonstrated.

Table 1. The distribution of question types in the BioASQ 7b training data.

Type	Counts
Factoid	779
Yes/No	745
Summary	667
List	556

3.2. Architecture

The end to end Question Answering system is a knowledge-based system where the user asks a question and the system gives an answer back based on the article and document in the database. The implementation is done in 3 stages (**see** figure 3):

- The retriever stage exists to perform data preprocessing by parsing the question and document pair into the format that can be used by the BERT model.
- The BERT stage of this architecture exists as a fine tuned BERT model or in other cases as a generic BERT model. This is the core

of the solution which will train on our data in order to understand the subject.

- The Reader is the final step which consists of applying or predicting an answer from a given question using the trained BERT model.

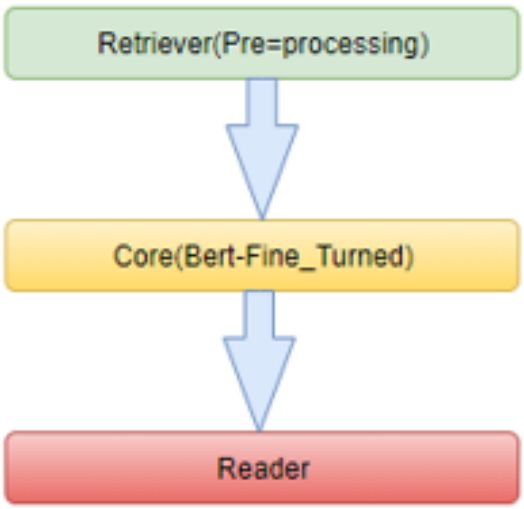

Figure 3. The overall architecture of the QA system has 3 major components.

3.2.1 Retriever (Pre-processing)

This retriever stage performs pre-processing, which is a very essential stage of the solution, it's consisting of converting the BioASQ dataset which is made of question and document to a SQuAD format dataset. This solution is based on BioASQ factoid type of question, as it shares a similar structure as SQUAD. As described in the dataset section, each entry in the BioASQ dataset is made of one question attached to multiple document URLs and context snippets. The model trains by using each individual document to attempt to answer the question. The same question is attempted across different documents, which increases the number of training repetitions for questions.

The documents that contained the necessary reference text were listed in the dataset as URLs. It was necessary for the model to use web scraping techniques to extract text from these URLs and transform them into a computer interpretable text format. The main source of these reference documents is from PubMed's website. The format necessary for model interpretation followed the format of the SQuAD QA challenge. The SQuAD format consists of reference passages and their respective question-answer sets. A reference passage is an article which contains answers or clues for answers and is denoted as the context in the dataset, each question-passage pair as an exact answer which may or may not exist in the passage. On retrieving the passage, a few different

approaches were used to extract the passage for each question.

Snippet as-is strategy: Using snippets in their original form is a basic method for filling passages. The starting positions of exact answers indicate the positional offsets of exact matching words. If a single snippet has more than one exact matching answer word, then the first answer is selected to form question-passage pair.

Full Abstract Strategy: In the Full Abstract Strategy, the model used the entire abstract, including the title of an article, as a reference passage. Full abstracts were retrieved from PubMed using their provided PubMed IDs. The extracted abstract is eventually searched to determine the location of the correct answer.

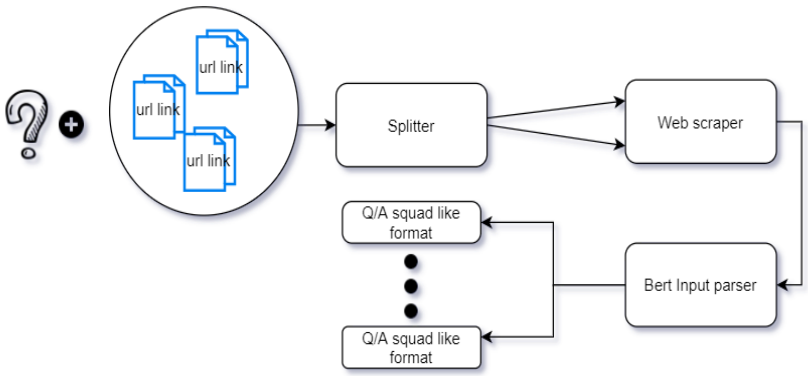

Figure 4. The Retriever function extracts data from the input documents and re-formats it for the BERT model

3.2.2 Core or Bert-Fine Tuned

This study uses the BERT model, which is a Bidirectional Encoder Representation from Transformers. This version of the transformer model stacks encoder attention mechanisms on top of each other. In summary, the BERT model is a language representation model that operates by training deep bidirectional representations from unlabeled text by conditioning on both left and right context in all layers. The left and right hand context along with attention features make the BERT model state of the art in dealing with language comprehension problems like QA [3].

The pre-trained BERT model can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task specific architecture modifications[3].

BERT pretraining consists of two phases: Masked Language Modeling (MLM) in which BERT intakes sentences with random words filled with masks. The BERT model

tries to output the masked tokens. In doing so, BERT is learning context between word tokens.

Next Sentence Prediction (NSP) follows, in which BERT intakes 2 sentences and determines whether sentence B follows sentence A. This segment helps BERT model understand context across different sentences.

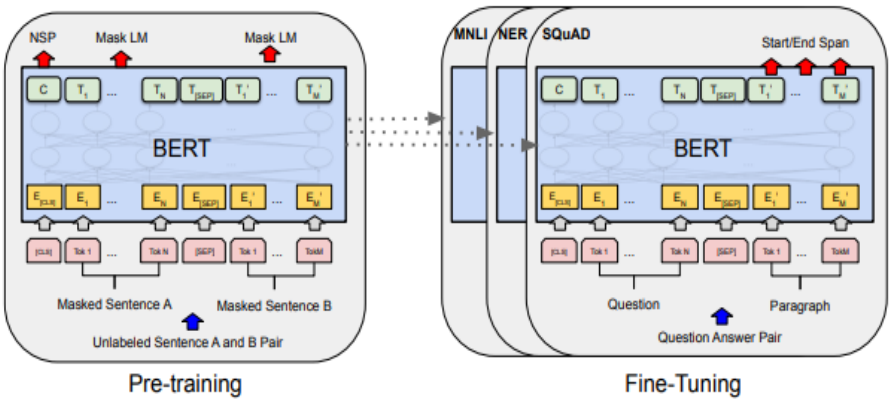

Figure 5. BERT pre-training and fine-tuning representation. [15]

During pre-training, BERT transforms the input text into token embeddings, segment embeddings, and positional encodings. Once trained, this study's BERT model was fine-tuned by adding an output layer for producing answers to questions.

Doing so, the BERT model can now process the tokens in the question and searches the corpus of documents using the question tokens. BERT uses special tokens to differentiate between question and answer segments. When given a question, the BERT model will analyze its own trained embeddings in search of the answer.

BERT will attempt to highlight a span of text containing the answer. This is represented as simply predicting which token marks the start of the answer, and which token marks the end.

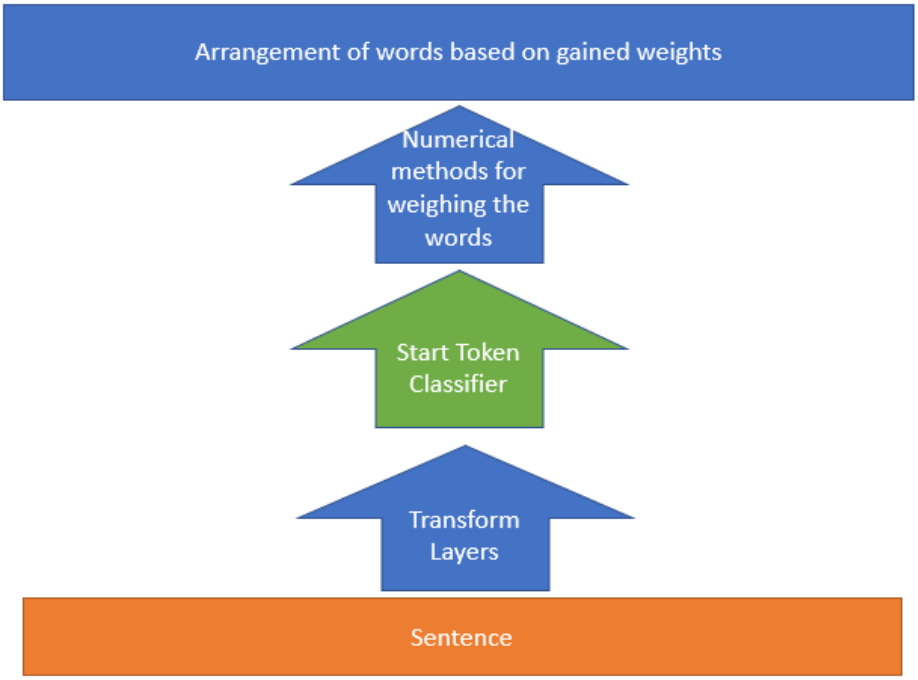

Figure 6. BERT start token prediction

For every token in the text, it was fed its final embedding into the start token classifier. The start token classifier only has a single set of weights (represented by the blue 'start' rectangle in the above illustration) which it applies to every word. After taking the dot product between the output embeddings and the 'start' weights, the SoftMax activation was applied to produce a probability distribution over all the words. Whichever word has the highest probability of being the start token is the one that BERT picks. This process is repeated using a separate weight vector in order to identify the end token.

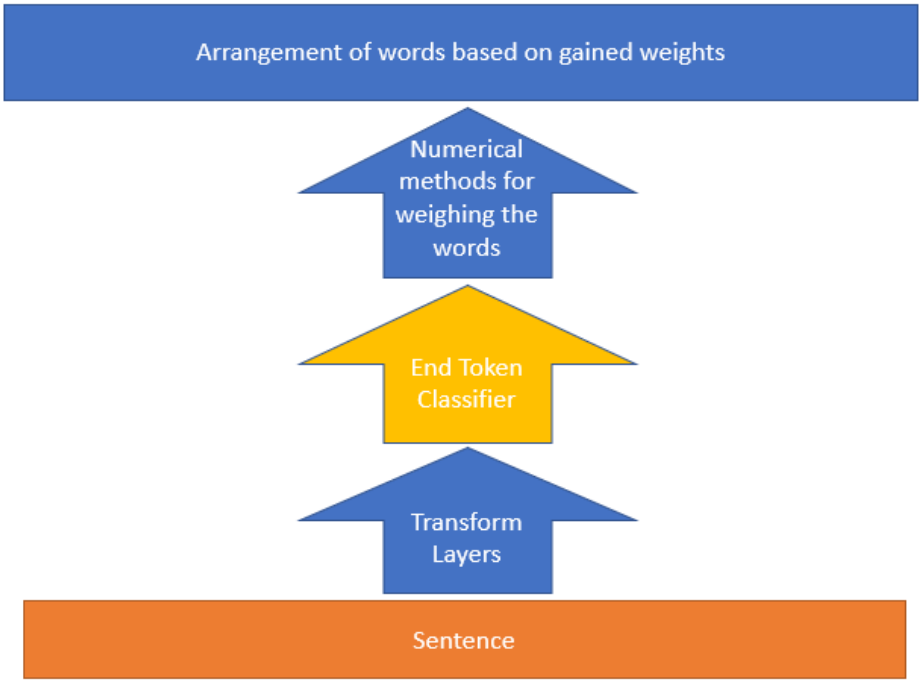

Figure 7. BERT end token prediction

After the data was formatted, the next step was to send it to the BERT model, so that the base BERT model can train on the dataset. BERT uses stacked encoder attention mechanisms to calculate weights for each input word and also performs two pre-training tasks to further gain comprehension of the BioASQ data.

Figure 8. BERT training section of QA architecture

3.2.3 Reader

The reader is the final segment of the QA solution which takes the given question already parsed into SQuAD format and uses the newly trained BioASQ-BERT model to read the question's passage and predict the relevant answer. Since the reader duplicates the question across different passages, it produced multiple answer per question/passage. The final answer was randomly selected from the list of answer.

Figures 9. The Reader segment is where the model uses its training to predict an answer

3.3 Experimental setups

BERT model uses some tensor flow features and the related packages were used to load the model and data set. BioASQ context documents were preprocessed by parsing the JSON data into the necessary format. Using BERT tokenizer features to format the input and inserting special tokens (cls), (SEP) at the begging of send of each sentence in order to tokenize the sentence as it was shown in figure 6,7. Sentence length and attention mask has been executed and placed with maximum sentence length of 128 .

(MAX_LEN = 128 --> Training epochs took ~5:28 each.)

Finally taking prepping steps and tokenizing the dataset made it through training and validation. "BertForSequenceClassification" was used for classification in continuing steps for training the model. As per feeding input data, the entire pre-trained BERT model and the additional untrained classification layer is trained on specific task.

4.0 Results

Here in Table 2 are the results of F1 score on four different run of BERT model on the dataset with four different approach on dataset to be trained. The best score was an F1 score of 76.44 and the model was able to generate correct answers to specific

questions. In the fourth approach of dataset training on BERT only contents of full abstracts were considered for training and evaluation and it shows great increase of F1 score.

Table 2: Performance results given different preprocessing methods. Scores on BioASQ 7b dataset.

Data Used for Training	Approach	F1 Score
BioASQ	Full Dataset with including "Impossible Answer"	10.88
BioASQ	Uses snippets of abstracts	23.67
BioASQ	Full Abstract without "Impossible Question" feature	65.29
BioASQ + SQuAD	Full Abstract without "Impossible Question" feature	76.44

As the table demonstrates, the initial approach provided a weak F1 score. This result was improved after abstract snippets from the reference texts were extracted and used. However, results at this point were still poor. After extracting full abstracts, the model greatly improved and began to show modest ability to answer questions. After including an additional QA dataset from SQuAD in training, results were improved once again. These results demonstrate how impactful data preparation was for our BERT QA model.

In Table 3 below, there are two examples of questions inputs and answers outputs that the fourth model was able to generate based on BioASQ 7b.

Table 3: Example question input and answer output from the QA Model.

No.	Type	Description
1	Question	"What is the mode of inheritance of Wilson's disease?"
	Context Text (Full abstract)	http://www.ncbi.nlm.nih.gov/pubmed/838566 In a survey in Israel of 50 patients with Wilson's disease, it was found that this disease occurred in all ethnic groups. In the Arab patients there was a significantly early age of onset and the disease followed a more severe course than that in the Jewish patients. The overall sex ratio of patients was nearly 1:1, and genetic analysis of 20 families confirmed an autosomal recessive mode of inheritance. The very similar age of onset and type of disease within sibships and the varying ages of onset noted between the Arab and Jewish patients suggest that the disease is genetically heterogeneous.
	Is Impossible	False
	Exact Answer	"autosomal recessive"
	Predicted Answer	"autosomal recessive"
2	Question	"What kind of chromatography is HILIC?"
	Context Text (Full abstract)	http://www.ncbi.nlm.nih.gov/pubmed/22946920 In bioanalysis, phospholipids may affect the precision and accuracy of LC-MS/MS methods and compromise the quality of the results, especially when samples in complex biomatrices are extracted by protein precipitation techniques
	Is Impossible	True
	Exact Answer	Hydrophilic Interaction Chromatography
	Predicted Answer	null

The results indicate decently effective performance on BioASQ with an F1 score of 76.44. Results were obtained by using the evaluating model performance against the BioASQ 7b test dataset, which contained questions related to our training data. This study's findings showed that BioASQ performance improved by an F1 score difference of ~11 with the inclusion of SQuAD training data along with the BioASQ training data. The findings showed that preprocessing the training data differently greatly affected results. Notably, this study experimented with what to include in the reference data to a question. The results showed that using the whole dataset content might not be efficient and due to redundancy of contents. So, with this approach the performance resulted very well as we only selected the full abstract contents to be read for BERT training. Using the full content of reference text abstracts was very helpful to increase the F1 score in comparison of 'Snipped content' and other parts of datasets. Considering whole content of dataset was considerably time consuming and did not show improvement of F1 score. Extracting reference passages effectively was essential to

improving performance during training. Performance also improved after removing impossible questions from the dataset.

5.0 Discussion

This study has shown that the BERT model and in general NLP methods have the potential to quickly understand massive volumes of text information while providing answers to related questions. The gain of information has always been integral in moving individuals, groups, and civilizations forward. That is why it is so valuable when a machine language method can spread massive volumes of accurate information while almost completely minimizing the cost of time or human input required to gain that information. Although the internet efficiently meets a large portion of society's information demand, there are endless scenarios where individuals and groups will benefit from quickly learning less generally available information. Until the development of machine QA solutions, gaining such information has always required considerable time invested in studying or the cost of employing a human teacher. The general limitation of machine QA is that it can only offer existing information at less than perfect accuracy.

This study approached the BioASQ challenge in a way that was advantageous for its resulting accuracy. Multiple data preprocessing approaches were experimented with in order to find a way to best train the QA model. Parsing the full abstract from the context documents gave the model the most information to work with. Removing is_impossible questions improved performance. Using the highly formatted dataset from the SQuAD QA challenge improved our model performance on BioASQ questions by 10%.

In this study, only factoid questions were tested. Yes/No, summary, is_impossible, and list questions did not achieve desirable accuracy under this approach. Before removing is_impossible questions, the majority of the questions were impossible. The proportion of questions that were possible and used to train the model was less than half. In the future, a more complex approach would be necessary to improve performance based on this dataset limitation.

Ethical implications of this research should be considered. When individuals genuinely need expert medical answers, artificial QA systems cannot be a replacement for actual expertise. It is the responsibility of a question's asker to find an appropriate answer source for their personal needs. This methodology should not be used to answer questions regarding personal or private information. Distribution of sensitive or private information could be considered illegal.

The F1 score was used as the result criteria because it measures true positive rate. In the field of QA, true positive rate is an effective measurement at determining how well the model is answering questions.

6 Conclusion

In this study, a BERT based QA system for the BioASQ biomedical question answering challenge was proposed. Because the size of the BioASQ question answer dataset is relatively small, numerous dataset preprocessing experiments were run to achieve best

results. The proposed model achieved an F1 score of 76.44 on factoid questions in the 7th BioASQ challenge. These results are comparable to the top performers in the challenge.

References

1. Vargas-Vera, M., Lytras, M., & Vargas-Vera, M. (2010). AQUA: A Closed-Domain Question Answering System. Information Systems Management, 27(3), 217–225. https://doi.org/10.1080/10580530.2010.493825
2. Buck, C., Bulian, J., Ciaramita, M., Gajewski, W., Gesmundo, A., Houlsby, N., & Wang, W. (2017). Ask the Right Questions: Active Question Reformulation with Reinforcement Learning.
3. Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
4. Bhutani, N., Jagadish, H., Mei, Q., Cafarella, M., Lasecki, W., Li, Y., & Mihalcea, R. (2019). Answering Complex Questions Using Curated and Extracted Knowledge Bases [ProQuest Dissertations Publishing]. http://search.proquest.com/docview/2355996900/
5. Tapeh, A., & Rahgozar, M. (2008). A knowledge-based question answering system for B2C eCommerce. Knowledge-Based Systems, 21(8), 946–950.
6. Pirtoaca, G., Rebedea, T., & Ruseti, S. (2018). Improving Retrieval-Based Question Answering with Deep Inference Models.
7. Seo, M., Kembhavi, A., Farhadi, A., & Hajishirzi, H. (2016). Bidirectional Attention Flow for Machine Comprehension.
8. Xiong, C., Zhong, V., & Socher, R. (2018). Dynamic Coattention Networks For Question Answering. arXiv.org.
9. Yang, W., Xie, Y., Lin, A., Li, X., Tan, L., Xiong, K., Li, M., & Lin, J. (2019). End-to-End Open-Domain Question Answering with BERTserini. arXiv.org.
10. Malik, N., Sharan, A., Biswas, P., & Malik, N. (2013). Domain knowledge enriched framework for restricted domain question answering system. IEEE Conferences,
11. Maria Vargas-Vera and Miltiadis D. Lytras ,AQUA: A Closed-Domain Question Answering System,Information Systems Management, 27:217–225, 2010.
12. Gupta, P., & Gupta, V. (2012). A Survey of Text Question Answering Techniques. IJCA, 53(4), 1–8.
13. Zhong, V., & Socher, R. (2017). Dynamic Coattention Networks For Question Answering. ICLR, 1–14
14.Sukhbaatar, S., Szlam, A., Weston, J., & Fergus, R. (2015). End-to-end memory networks. *Advances in Neural Information Processing Systems*, *2015-Janua*, 2440–2448.
15. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *2017-Decem*(Nips), 5999–6009.
16. Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).
17. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.
18. Cho, v. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation.