BERT based pre-trained models for Query-focused Extractive Summarization of Biomedical Evidence

Anonymous ACL submission

Abstract

Evidence from published biomedical papers has been expanding enormously in recent years, and handling such an interminable amount of information is challenging. Using this biomedical evidence efficiently can assist medical specialists in quick diagnosis and treatment. Text Summarization is a technique used to obtain relevant and essential information from text quickly and efficiently. BERT based models have achieved state-of-the-art results in a wide range of text applications including question answering and text summarization. This paper discusses how a straightforward use of BERT based pre-trained models outperforms most of the systems participating in the BioASQ8b shared task for the goal of query-focused summarization. In particular, a system using the Bert-base pre-trained model exhibited the highest ROUGE-SU4 and ROUGE-2 recall scores in three out of five batches and close scores to the highest ones in two batches of BioASQ 8b run challenge submissions.

1 Introduction

The amount of data in text, images, sources on the web etc., is increasing each year. To handle this massive growth of information, the field of Human Language Technologies (HLT) aims at developing computer systems that can process and interpret text documents. HLT involves a wide range of interdisciplinary activities to enable people to communicate with the machines through natural language or natural communication skills efficiently (Bird et al., 1997). Some of the well-known intelligent applications in HLT are Information Retrieval (IR), Text Classification (TC), Question Answering (QA) and Text Summarization (TS).

Question Answering is a method of providing precise answers to a given question in Natural language by accessing a collection of documents or a database related to the question (Strzalkowski and Harabagiu, 2006). *Text Summarization* is the method of producing a summary of the whole or required parts of the document by selecting and condensing the relevant and essential information only. These summaries act as a surrogate for the original document (Sparck-Jones, 1999) and some of the common types of summaries are Abstractive and Extractive Summaries. Extractive Summaries are produced by extracting the relevant sentences from documents without any changes to the sentences, which might lack coherence and is prone to dangling anaphora. To overcome this issue, abstractive summaries are produced by extracting the relevant sentences and adding new text that provide a context to the summary while handling the repetition of information. Though Abstractive summaries can produce better quality summaries, the underlying complexities in the natural language make this task both challenging and complex (Gupta and Gupta, 2019).

With the enormous growth of information in the Biomedical domain, it is imperative to develop systems that can assist specialists in quickly diagnosing and treating disease through efficient access to the information. The increase in the number of life-threatening diseases identified daily motivates to improve the facilities of the Health care sector using technology for a better future. BioASQ [1] is an organization that focuses on developing intelligent applications that can assist in the Biomedical Domain. It is a benchmark for some of the best systems submitted from recognized research universities and industry professionals and hence considered a basis for this research. BioASQ Task B challenge focuses on question answering including the generation of paragraph-sized answers called “ideal answers”. This paper discusses the use of

[1] http://www.bioasq.org/

query-based extractive summarization techniques for the generation of ideal answers, with a primary focus on the Bi-directional Encoder Representation of Transformers (BERT) model (Devlin et al., 2018). BERT based pre-trained models are used to propose a system that minimizes the complexity of architecture and reduces the computational resources. This paper explains these experiments, followed by elucidating the better or as-good results obtained by the systems proposed compared to the current best systems participating in the BioASQ 2020 (8b run) challenge.

2 Background and Related Work

Text Summarization can be combined with Question Answering tasks to reduce the human processing time by providing them summaries, thus eliminating the necessity of dealing with the whole documents when answering queries. The summaries representing the critical information in a document assist in better performance of the systems. However, the key challenge when producing automated summaries is to generate summaries that approximate human-generated summaries in various factors affecting the type and quality of human summaries like personal opinions, person-oriented paraphrasing, and varied outlook of emphasized information by different people. There has been extensive research in working with the integration of Question Answering, Summarization. BioSquash (Shi et al., 2007) is a medical domain system based on a generic summarizer, which aims at answering a question by summarizing multiple biomedical documents. It involves four components: Annotator, Concept Similarity, Extractor, and Editor modules that perform individual tasks to generate a final summary. QAAS (Torres-Moreno et al., 2009) is also based on a generic multi-document summarizer in which multiple compression rates are coupled with the QA system to reduce the summary space, thus improving the number of correct answers obtained. While TS can be used for QA tasks, the inverse approaches, i.e. using QA for TS, are also observed. In these approaches, the QA system determines the importance of the sentences, using scores and a set of queries, whose output is integrated into a generic multi-document summarizer to generate the final summary (Mori et al., 2005).

Query-based Extractive Summarization is the process of generating answers by extracting the relevant sentences related to a query from a given text document or set of sentences without paraphrasing or adding additional words. Various approaches using Deep Learning and Reinforcement Learning frameworks based on Regression and Classification setups have been proposed. These approaches involved using pre-defined language models ranging from simple word2vectors, tf-idf to variations of twin networks, LSTM, BERT, BioBERT etc. Among these approaches, BERT based architectures displayed state-of-the-art results for query-based summarization tasks, as we will see below. These systems are evaluated using performance metrics like Recall, Precision, Accuracy, F1 scores and ROUGE scores. ROUGE scores determine the closeness of generated summaries to human-generated summaries, hence they are used in this paper. On the light of the previous remarkable state-of-the-art results using BERT, we investigate the straightforward use of BERT pre-trained models. The system involves an inverse fashion implementation, i.e. using Question and Answers to score sentences followed by generating the final summary. The data provided by BioASQ involves four types of answers to the questions: answers in the form of Yes/No, List, Facts known as Exact Answers and answer as a Summary known as Ideal answer. The system proposed in this paper focuses on answers as a summary, i.e. generating Ideal answers. Further in this Section, the BERT architecture and some of the best-proposed summarization systems based on BERT in the BioASQ challenge are discussed.

2.1 BERT Architecture

Bi-directional Encoder Representations from Transformers (BERT) is a language representation model that overcomes the unidirectional model limitations using a Masked Language Model (MLM). In this model, some tokens are hidden (“masked”) and the goal is to predict their word embedding based on the left and right side words context (Devlin et al., 2018). Consider the example in Figure 1, where the word ‘ball’ has two different meanings (known as Homonyms) — An event and a sports ball. BERT model provides the embedding of the word ‘ball’ by evaluating the context of the right and left side words.

The inputs in the form of tokens are sent to the BERT model to obtain contextual embeddings of the words in the sentence, which are further used

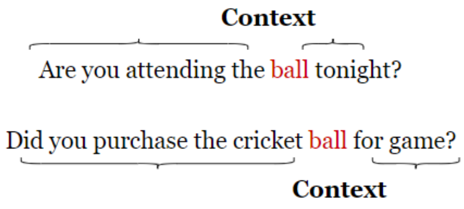

Context
Are you attending the ball tonight?
Did you purchase the cricket ball for game?
Context

Figure 1: Example of Homonyms

on the downstream tasks as required. The diagrammatic overview of how the inputs are given to the complex BERT model to obtain contextual embeddings can be seen in Figure 2. The classification token '[CLS]' added at the beginning of each input sentence is a sentence vector that acts as a sentence representative used in Classification and Next sentence prediction tasks.

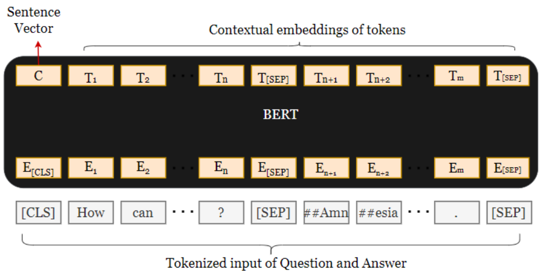


Figure 2: Overview of input and output in BERT model (Han and Tsai, 2020)

The BERT architecture is complex. The black box of the BERT model architecture involves a series of encoder representations from transformers. Each encoder layer takes input from the previous encoder layer and outputs an intermediate representation of the word. Each encoder layer has an attention layer followed by a feed-forward network. The attention layer assigns an embedding to the word by assessing its importance in the sentence. The self-attention process involves a series of mathematical calculations to compute the embeddings, which can be further studied in the paper by Alammar (2018). The number of encoder layers depends on the chosen model configuration, and the architecture details are further discussed in Section 3.

2.2 BERT based systems

This section discusses some of the best systems submitted in the BioASQ challenge that involved the BERT-based architecture, focusing on the 7b (2019) and 8b (2020) runs. These systems focus on generating ideal answers; however, some BERT based systems that produced good results for exact answers are also discussed. The system *'Bio-AnswerFinder'* proposed by UCSD involved three processing phases, namely, Question, Document and the Answering phases, each allocated to specific tasks. The sentences are ranked after the question focused filtering using a weighted-Relaxed Word Mover's Distance (wRWMD) (Kusner et al., 2015) similarity, to perform clustering of the word/sentence embeddings in an optimized way. Re-ranking the sentences using fine-tuned BERT and BioBERT classifier were also experimented to obtain the final summary. Though BERT based rankings produced better results, no different results were observed using BioBERT (Ozyurt et al., 2020). BioBERT, fine-tuned on the SQUAD dataset, yielded noticeable results to obtain exact answers.

The system named *'DMIS'* proposed in the eighth edition of BioASQ is based on the proven improvement in the performance of QA when the learning relationships between the sentence pairs are induced. This system used BioBERT to transfer the knowledge of Natural Language Inference (NLI) to biomedical Question Answering, generating good results for exact answers. An abstractive summarization model BART, combined with several pre-processing and post-processing steps to generate a final summary, is observed to produce good results (Jeong et al., 2020). The system *'pa'* proposed by ITMO University, Russia, displayed good performance using the baseline approaches compared to other suggested systems. It used the BM25 algorithm (Robertson and Zaragoza, 2009) for sentence retrieval followed by re-ranking using BERT variations fine-tuned on BioASQ documents. Other re-ranking methods using word2vec based on cosine similarity and BERT scores relevance produced good summaries (Kazaryan et al., 2020); however, an approach experimented using BioMed-RoBERTa (Gururangan et al., 2020) did not present good results.

The system proposed by *'DAIICT'* University, India, in the eighth edition of the BioASQ challenge, has shown good ROUGE-SU4 scores and high recall scores. This system focused on implementing query-graph based summarization techniques, including named-entity information due to their better performance in previous studies (Moradi and Ghadiri, 2018). Along with the use of standard graph-based techniques like LexRank (Erkan and Radev, 2004), and TextRank (Mihalcea and Tarau, 2004), UMLS based query-specific graphs and Query Sentence Matching (QSM) were also utilized to select sentences and obtain summary (Sankhavara and Majumder, 2020). This system did not utilize the BERT model; however, it obtained top scores in one of the submissions in BioASQ. The use of QSM without including UMLS showed better performance, showing the importance of ontological knowledge to obtain better summaries.

The system *'MQ'* proposed by Macquarie University used variants of BERT and BioBERT in regression and classification setups. The variants of BERT and BioBERT approaches involved experiments using different types of embedding generator and reductor to generate word and sentences embeddings, along with the inclusion of Bi-directional LSTM and siamese-BERT based architectures. The inclusion of LSTM as a reductor in BERT and BioBERT based models produced better results and stood as one of the best systems in BioASQ submissions (Molla et al., 2020, 2021).

The system *'sBERT'* in the eighth edition of BioASQ used a multi-task learning system based on classification and regression setups using a few MQ system methods for data pre-processing. The experiments involved different embedding models like BioBERT-NLI — A BioBERT model fine-tuned on SNLI and MultiNLI dataset followed by either classification or regression layers (Nentidis et al., 2020). The system *'NCU-IISR'* experimented with approaches using BioBERT fine-tuned on SQUAD dataset for ranking the sentences. This system also used a logistic regression model inspired by the MQ system to predict the similarity between the questions and answers, thus predicting the final summary by extracting top 'n' sentences. This model used the RELU activation layer and MSE loss function while fitting the model (Han and Tsai, 2020; Zhang et al., 2021).

Analysis

Based on the previous systems discussed, a few key observations put together to produce better summaries are:

i. Classification approaches perform better than Regression approaches.

ii. Transfer Learning, i.e. using pre-trained language models obtained better results.

iii. BERT and Bio-BERT based model architectures in combination with other summarizing algorithms obtained state-of-the-art results.

Though the previous systems proposed utilized BERT and BioBERT with external algorithms, there is a possibility that BERT based pre-defined models can produce as good or better results with simpler architectures and with less computing resources required. This led to working on the research question "Can simple BERT based predefined architectures obtain better results?". This can also be framed as "Is the complexity of the model hindering the better performance of the model?" — however, it is to be noted that BERT by itself has a complex architecture. Thus, this paper proposes a new system based on pre-defined BERT based models inspired by the 'MQ' and 'NCU-IISU' systems.

3 Methodology

This Section discusses the methods used to build the proposed system explaining the details of the BERT based pre-trained models. This research work is based on the use of BertForSequenceClassification, a pre-defined model provided by Hugging Face's[2] Transformers.

BertForSequenceClassification

The BertForSequenceClassification model involves a classification/regression head and can be used for either of the tasks. The proposed model involves using a classification head, and the model's architecture is based on the pre-trained model chosen. The model begins by taking word and token type embedding as inputs, along with the position embeddings. The essential inputs required for the model are as follows.

1. *Word indices / Input ID's:* The tokens obtained after tokenizing are converted into a numerical format for the machine to understand, known as Word indices/Input ID embeddings.

2. *Attention masks:* Masking is performed on the words to differentiate the actual word embeddings from padded embeddings. Attention masks are in the form of 1's and 0's, with '1' representing the actual word and '0' representing the padded words. This way, the words that do not have meaning are not given attention.

[2] https://huggingface.co/transformers/

3. *Segment ID's / Token type ID's:* Segment ID's are used to differentiate the first sentence from a sentence following it, assisting the model to learn the difference between the input sentences. These are in alternating sequences of 0 and 1, changing when a new input sentence begins.

4. *Position embeddings:* These are the sequential numbers given to the tokens for determining their position in the given input sentence.

The architecture of the BertForSequenceClassification model can be seen in Figure 3 with the details of the inputs and working of the model architecture discussed further below.

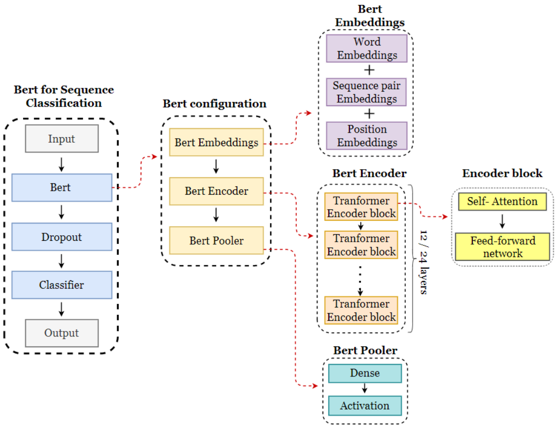


Figure 3: Internal working of BertForSequenceClassification model

The BertForSequenceClassification [3] model begins with a BERT layer that takes inputs to obtain contextual embeddings, followed by a Dropout layer that assists in adding noise to the network in order to reduce over-fitting of the model and ends with a final classifier layer to categorize/score the sentences as required. The three layers in the BERT configuration have different functionalities. The embeddings are processed in the embedding layer and forwarded to the series of encoder layers with an attention layer inside each encoder. BERT Pooler performs pooling using the Tanh() activation function, which provides outputs in the range $(-1, 1)$. The number of Transformer encoder blocks depends on the chosen pre-trained model for the task. With BERT obtaining state-of-the-art results as discussed previously, the pre-trained models based on BERT are chosen for the experiments and are further explained.

Pre-trained models

The HuggingFace transformers library provides pre-trained models based on BERT. These models can be imported and fine-tuned on task-specific data. Along with BERT based pre-trained models, DistilBERT (A distilled version of BERT) and BioBERT (A BERT model fine-tuned on vast corpora of biomedical documents) have been experimented. BioBERT model can be termed as a domain-specific language representation model due to its focus on Biomedical information. Each of these models has a base and a large version, which vary in architecture. The SequenceClassification library changes accordingly based on the chosen BERT based model and these details are shown in Table 1.

Transformer Libraries	**Pre-trained model**	**Architecture**
Bert For Sequence Classification	Bert-base-uncased	12 transformer encoder layers, 768-hidden layers, 12 attention heads, 110 million parameters
	Bert-large-uncased	24 transformer encoder layers , 1024-hidden layers, 16 attention heads, 336 million parameters
AutoModel For Sequence Classification	Biobert-base-cased	12 transformer encoder layers, 768-hidden layers, 12 attention heads, 110 million parameters
	Biobert-large-cased	24 transformer encoder layers , 1024-hidden layers, 16 attention heads, 336 million parameters
Distilbert For Sequence Classification	Distilbert-base	6 transformer encoder layers , 768-hidden layers, 12 attention heads, 66 million parameters

Table 1: BERT based pre-defined models architectures used in the experiments

3.1 Methods or Model Description

HuggingFace [4] Transformers is a software built on PyTorch and TensorFlow libraries that provides various pre-defined models trained on a huge text corpus. PyTorch Transformers is a library built on PyTorch and provides a diverse range of pre-trained models and pre-trained weights that can be imported and fine-tuned on the required data. This method of fine-tuning on task-specific data is termed Transfer learning and can be customized based on the specific task.

Data Collection and Pre-processing

BioASQ provides biomedical text each year, and the dataset provided in 8b run (2020) is chosen for the experiments, along with the five batches of test data provided in each run. We chose to use the dataset from the 8b run, and not the more recent 9b run, so that we can compare our results against those of participating systems. At the time of conducting our experiments, it was not possible

[3] https://huggingface.co/transformers/model_doc/bert.html#bertforsequenceclassification

[4] https://huggingface.co/transformers/

to do this with the 9b data set. The data provided is in JavaScript Object Notation (JSON) format and needs to be pre-processed before training the model. The data consists of a list of questions with multiple snippets, i.e. answer texts and additional details related to questions like question ID, type etc. This data in JSON format is processed to extract the questions and answers (candidate sentences) to provide input to the model. The labels (i.e. ground truth) are determined based on ROUGE-SU4 scores, a metric used to measure the model's performance by comparing the sentence against human-generated summaries. The scores of the answer snippets for each question are thus calculated, and the top five answer snippets with the highest scores are labelled '1', while the remaining answer snippets are labelled '0' for training. 'Bert Tokenizer' built using Word-piece algorithm is used in the experiments (Eram Munawwar, 2020) to split the input text into sub-words used as tokens which are passed into the model as embeddings. An example of tokenizing a sentence using BERT tokenizer can be seen in Figure 4.

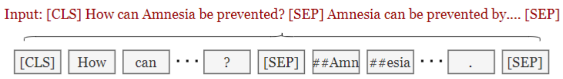

Input: [CLS] How can Amnesia be prevented? [SEP] Amnesia can be prevented by.... [SEP]
[CLS]
How
can
· · ·
?
[SEP]
##Amn
##esia
· · ·
.
[SEP]

Figure 4: Tokenizing sentence using BERT tokenizer based on Word-piece algorithm

Model details

As stated previously, BERT, DistilBERT and BioBERT based models are used in the experiments. The general architecture of the model using BERT is shown in Figure 5, which is similar to the experiments using DistilBERT and BioBERT.

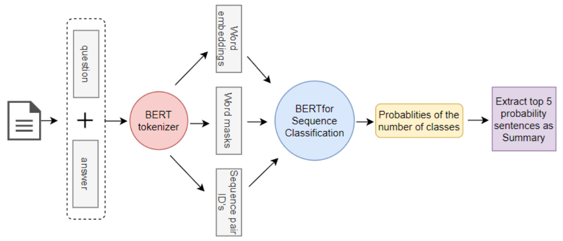

question
+
answer
BERT tokenizer
Word embeddings
Word masks
Sequence pair ID's
BERTfor Sequence Classification
Probablities of the number of classes
Extract top 5 probability sentences as Summary

Figure 5: Architecture of the model proposed in this paper

The model is trained with 80% of the data, while the remaining is used as Validation data to assess the model's performance on unseen data. The input in the form of a unique string of question and the respective answer snippet are combined before being sent to the BERT tokenizer. The question and answer snippet are joined using special tokens '[CLS]' at the beginning and '[SEP]' between the question, answer and end of the text. These inputs are sent to the BERT tokenizer, where the sentences are broken down into tokens that are processed to obtain word embeddings, mask embeddings, and sequence ID embeddings. These input embeddings are converted into tensors and sent to the BertForSequenceClassification transformer model for training. It is to be noted that the BERT layers of the model are frozen during training, i.e. back-propagation to find the best weights is disabled for BERT layers, and the weights of the final classifier layer are only fine-tuned. The parameters set to fine-tune the model are shown in Table 2.

Parameter	**Description**	**Value**
Epochs	Indicates the number of times the models needs to run over the entire given dataset.	4
Batch	Indicates the number of divisions a dataset needs to be split into in one epoch.	32
Learning rate	Indicates the rate of change in the weights of the neural network based on the loss obtained.	$2e-5$
Number of warm-up steps	Indicates the number of steps until which the learning rate reaches a low value a '0, before the scheduling function starts.	100
Number of total steps	Indicates the total number of steps in the training phase of the model.	1000

Table 2: Parameters used for training the model

The loss function chosen to back-propagate and update the weights is the 'CrossEntropyLoss' function. 'AdamW optimizer' is chosen for the experiments where the weight decay is fixed and is independent of the learning rate. A 'get linear schedule with warmup' scheduler is chosen for the system where the learning rate decreases from '2e-5' to '0' in the warm-up phase, followed by a linear increase to the defined learning rate from '0'. The number of warm-up steps is '100' among the '1000' total number of steps and are chosen based on the best parameters retrieved through separate research experiments results. The trained model to predict scores of sentences is validated on unseen data to evaluate the model's performance.

Once the model is trained and validated, the model is tested on Test data. BioASQ provides five batches of Testing data, which is pre-processed and provided as inputs to the model, similar to the training phase. The trained model predicts the scores of the sentences. During this phase, the code can be structured based on what is required in the task. The system proposed in this paper returns the top five sentences with maximum scores for each question as a summary, by implementing a set of steps as shown below.

1. Each time, the answer snippets of a unique question are sent separately to the model by joining the question and each answer.

2. The logits/prediction probabilities obtained for each sentence are sorted to obtain the indices of the top 5 sentences with maximum score. If the number of sentences is less than five, all the sentences are returned as a summary.

Evaluation

The evaluation metrics chosen in these experiments are Accuracy and F1 score due to their suitability to understand the proposed model's performance. Besides, BioASQ provides ROUGE-2 and ROUGE-SU4 recall scores[5] by comparing the generated summary with the human-generated summaries to assess the proposed model's performance. Further, the results obtained through these experiments are discussed in Section 4.

4 Results and Discussion

The results obtained by the systems proposed using BERT based predefined models are evaluated using Accuracy and F1 score with the scores shown in Table 3. Due to the lower scores generated using the bert-large model, Distilbert-large model has not been experimented.

Model	Accuracy	F1-Score
Bert-base	**0.7060**	**0.5963**
Bert-large	0.7055	0.5958
Distilbert-base	0.7056	0.5961
Biobert-base	0.7057	0.5962
Biobert-large	0.7055	0.5960

Table 3: Accuracy and F1-score of BERT pre-trained models

Based on the results obtained, it can be observed that the Bert-base model obtained better results, while the next best results were by Biobert-base. These models have been experimented on the test data provided by BioASQ and submitted to the leaderboard of BioASQ. The ROUGE-2 and ROUGE-SU4 recall scores obtained for these systems are as shown in Tables 4 and 5.

It can be observed that the results obtained by Bert-base are higher compared to other experiments, followed by the scores of BioBERT-base. Based on these results, a mean and standard deviation of the model's ROUGE-2 and ROUGE-SU4 recall scores obtained in five batches is calculated. These values can be observed in Table 6 along with the error bar graphs as shown in Figures 6 and 7.

Model	ROUGE-2 Recall				
	Batch-1	Batch-2	Batch-3	Batch-4	Batch-5
Bert-base	**0.6488**	**0.5843**	**0.5467**	**0.5121**	**0.6258**
Bert-large	0.5884	0.5775	0.5044	0.4757	0.569
Distilbert-base	0.596	0.5442	0.5052	0.5054	0.5849
Biobert-base	0.6221	0.5657	0.4995	0.4841	0.5936
Biobert-large	0.5838	0.5532	0.5122	0.5094	0.5771

Table 4: ROUGE-2 recall scores of proposed systems in 5 batches

Model	ROUGE-SU4 Recall				
	Batch-1	Batch-2	Batch-3	Batch-4	Batch-5
Bert-base	**0.6529**	**0.5927**	**0.5586**	**0.5223**	**0.62679**
Bert-large	0.5986	0.5869	0.5216	0.4884	0.5737
Distilbert-base	0.6076	0.5508	0.5223	0.5139	0.591
Biobert-base	0.6261	0.5727	0.5165	0.4954	0.596
Biobert-large	0.5896	0.5589	0.5306	0.5226	0.5816

Table 5: ROUGE-SU4 recall scores of proposed systems in 5 batches

Model	Mean ± StDev (ROUGE 2 Recall)	Mean ± StDev (ROUGE SU4 Recall)
Bert-base	$\mathbf{0.5835 \pm 0.0559}$	$\mathbf{0.5907 \pm 0.0522}$
Bert-large	0.5430 ± 0.0499	0.5538 ± 0.0469
Biobert-base	0.5530 ± 0.0596	0.5613 ± 0.0545
Biobert-large	0.5471 ± 0.0351	0.5567 ± 0.0298
Distilbert-base	0.5471 ± 0.0428	0.5571 ± 0.0413

Table 6: Mean ± Standard Deviation of the ROUGE-2 and ROUGE-SU4 recall scores obtained by models in 5 batches

It can be observed that *Mean ± Standard Deviation* of the bert-base model outperformed the other models, and biobert-base model was second best. We can also observe that the Biobert base and large model scores have nearly close confidence intervals but vary noticeably compared to the score interval of the bert-base model in every batch. Even though the confidence intervals overlap, the bert-base model was best in all the batches, which gives reassurance of the higher performance of this model.

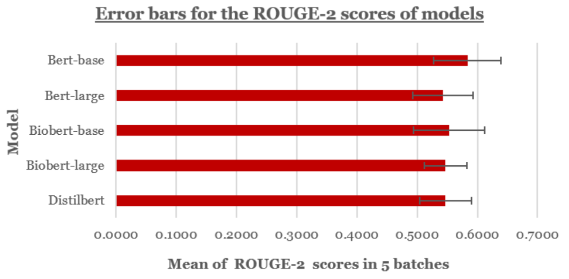


Figure 6: Error bar graph for ROUGE-2 scores of the models

[5] BioASQ did not provide ROUGE precision or F1 scores.

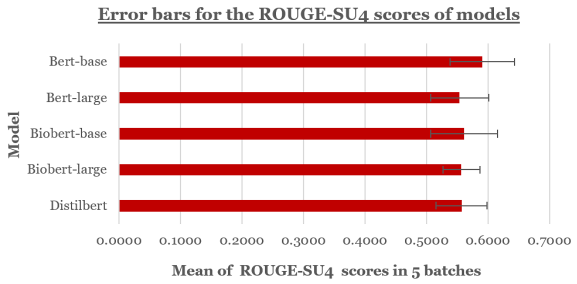
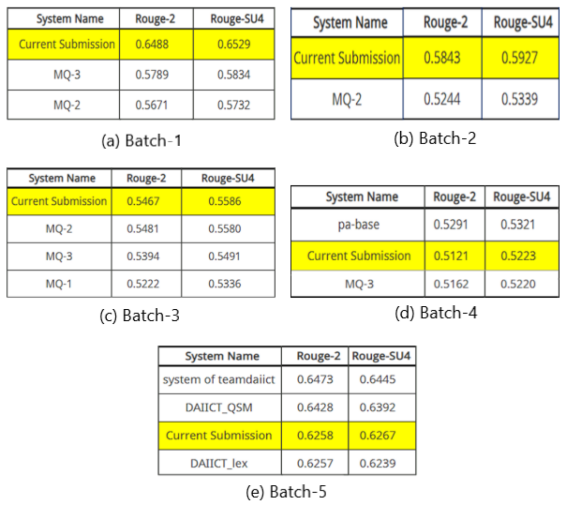


Figure 7: Error bar graph for ROUGE-SU4 scores of the models

Discussion

Based on these results, the BERT-based system's performance showed better scores using all the evaluation metrics. Along with this, BioASQ provides a leaderboard that displays the system scores comparing with other submitted systems. The system proposed in this paper stood as the top system in three batches among five, with results close to the top system in the other two batches. These leaderboard scores are shown in Figure 8.



System Name	Rouge-2	Rouge-SU4
Current Submission	0.6488	0.6529
MQ-3	0.5789	0.5834
MQ-2	0.5671	0.5732

(a) Batch-1

System Name	Rouge-2	Rouge-SU4
Current Submission	0.5843	0.5927
MQ-2	0.5244	0.5339

(b) Batch-2

System Name	Rouge-2	Rouge-SU4
Current Submission	0.5467	0.5586
MQ-2	0.5481	0.5580
MQ-3	0.5394	0.5491
MQ-1	0.5222	0.5336

(c) Batch-3

System Name	Rouge-2	Rouge-SU4
pa-base	0.5291	0.5321
Current Submission	0.5121	0.5223
MQ-3	0.5162	0.5220

(d) Batch-4

System Name	Rouge-2	Rouge-SU4
system of teamdaiict	0.6473	0.6445
DAIICT_QSM	0.6428	0.6392
Current Submission	0.6258	0.6267
DAIICT_lex	0.6257	0.6239

(e) Batch-5

Figure 8: Leader-board of BioASQ submissions

From these results, it can be stated that the system proposed using the bert-base produced better results in most of the batches. It is to be noted that the hyper-parameters chosen for BioBERT are not the best ones, and experimenting with these may yield better results than the current ones. Looking at other systems with slightly higher scores than the proposed system, 'pa' used the BERT based re-ranking, which showed good results in a batch and the non-BERT method used by 'DAIICT' obtained better performance in another batch which can be due to the inclusion of ontological knowledge using UMLS. However, it can be observed that the difference is relatively minimal when compared to a greater score difference of the proposed system in other batches. These results show that the proposed system using pre-trained BERT models with minimal computing resources provides summaries with greater similarity to human-generated summaries.

5 Conclusion and Future Studies

Concluding this paper, the results show that BERT-based pre-defined models can produce some of the best results and this can be termed as an exciting finding and shows the effectiveness of BERT based models. BERT has a complex architecture with multiple layers, and each layer is known for its high functionalities. This concept is taken as the lead for this research of working with BERT based pre-defined architectures alone to minimize the complexity of the model. Along with this, pre-trained models reduced the requirement of vast computational resources to train the model.

Future Work

Fine-tuning Parameters: The parameters used in the experiments are the best provided by various published research works for the BERT fine-tuning tasks. Hence, the hyper-parameters of BioBERT based models can be experimented with to obtain better results. Time constraints made it considerably challenging to perform this task.

Question Type: Currently, the summary produced is a five-sentence summary which is irrespective of the type of the question. The data consists of question types such as "yesno" that can be answered with single line answers, and including information about the question type can improve the scores when evaluating with human-generated summaries.

Ontological Knowledge Inclusion and Abstractive Summarization Techniques: The inclusion of ontological knowledge in models is observed to produce better summaries (Sankhavara and Majumder, 2020). The ways of including this in the BERT models, along with exploring the abstractive summarization techniques using variations of BERT (Jeong et al., 2020), can be carried out as future work.

References

Jay Alammar. 2018. The illustrated transformer. *GitHub Blog, Online: http://jalammar. github. io/illustrated-transformer*.

Steven Bird, Branimir Boguraev, Martin Kay, David McDonald, Don Hindle, and Yorick Wilks. 1997. *Survey of the state of the art in human language technology*, volume 12. Cambridge university press.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Eram Munawwar. 2020. A comprehensive guide to subword tokenisers.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Som Gupta and SK Gupta. 2019. Abstractive summarization: An overview of the state of the art. *Expert Systems with Applications*, 121:49–65.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Jen-Chieh Han and Richard Tzong-Han Tsai. 2020. NCU-IISR: Using a pre-trained language model and logistic regression model for BioASQ task 8b phase b. In *CLEF 2020 Working Notes*.

Minbyul Jeong, Mujeen Sung, Gangwoo Kim, Donghyeon Kim, Wonjin Yoon, Jaehyo Yoo, and Jaewoo Kang. 2020. Transferability of natural language inference to biomedical question answering. *arXiv preprint arXiv:2007.00217*.

Ashot Kazaryan, Uladzislau Sazanovich, and Vladislav Belyaev. 2020. Transformer-based open domain biomedical question answering at BioASQ8 challenge. In *CLEF 2020 Working Notes*.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. PMLR.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Diego Molla, Christopher Jones, and Vincent Nguyen. 2020. Query focused multi-document summarisation of biomedical texts. In *CLEF 2020 Working Notes*.

Diego Molla, Urvashi Khanna, Dima Galat, Vincent Nguyen, and Maciej Rybinski. 2021. Query-focused extractive summarisation for finding ideal answers to biomedical and COVID-19 questions. In *CLEF2021 Working Notes*.

Milad Moradi and Nasser Ghadiri. 2018. Different approaches for identifying important concepts in probabilistic biomedical text summarization. *Artificial intelligence in medicine*, 84:101–116.

Tatsunori Mori, Masanori Nozawa, and Yoshiaki Asada. 2005. Multi-answer-focused multi-document summarization using a question-answering engine. *ACM Transactions on Asian Language Information Processing (TALIP)*, 4(3):305–320.

Anastasios Nentidis, Anastasia Krithara, Konstantinos Bougiatiotis, and Georgios Paliouras. 2020. Overview of BioASQ 8a and 8b: Results of the eighth edition of the BioASQ tasks a and b. In *CLEF Working Notes*.

Ibrahim Burak Ozyurt, Anita Bandrowski, and Jeffrey S Grethe. 2020. Bio-answerfinder: a system to find answers to questions from biomedical texts. *Database*, 2020.

Stephen Robertson and Hugo Zaragoza. 2009. *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc.

Jainisha Sankhavara and Prasenjit Majumder. 2020. Query-focused biomedical text summarization in BioASQ 8b.

Zhongmin Shi, Gabor Melli, Yang Wang, Yudong Liu, Baohua Gu, Mehdi M Kashani, Anoop Sarkar, and Fred Popowich. 2007. Question answering summarization of multiple biomedical documents. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pages 284–295. Springer.

Karen Sparck-Jones. 1999. Automatic summarizing: Factors and directions," in" advances in automatic text summarization. *Evaluations*, pages 6–7.

Tomek Strzalkowski and Sanda Harabagiu. 2006. *Advances in open domain question answering*, volume 32. Springer Science & Business Media.

Juan-Manuel Torres-Moreno, Pier-Luc St-Onge, Michel Gagnon, Marc El-Beze, and Patrice Bellot. 2009. Automatic summarization system coupled with a question-answering system (QAAS). *arXiv preprint arXiv:0905.2990*.

Yu Zhang, Jen-Chieh Han, and Richard Tzong-Han Tsai. 2021. NCU-IISR/AS-GIS: Results of various pre-trained biomedical language models and linear regression model in BioASQ task 9b phase b. In *CLEF 2021 Working Notes*.