BioELECTRA: An Efficient Approach to Biomedical Question-Answering

Student Name: Molly Hayward

Supervisor Name: Dr Noura Al-Moubayed

Submitted as part of the degree of MEng Computer Science to the

Board of Examiners in the Department of Computer Sciences, Durham University

***Abstract* —**

Context / Background – Pretrained language models have demonstrated their efficacy in question answering (QA); however, more advanced hardware and computing resources are required to train these complex models. Due to the specialised vocabulary used by biomedical-corpora, recent advancements in biomedical QA have shown that domain-specific pretraining is necessary to achieve competitive results. This highlights the scope for applying an efficiently-trainable model to biomedical QA.

Aims – The aim of this project is to adapt ELECTRA, a transformer-based language model, for the biomedical-domain by harnessing its highly-efficient pretraining objective. We aim to investigate the extent to which pretraining on biomedical corpora improves downstream biomedical QA performance.

Method – We pretrain the ELECTRA-Small and ELECTRA-Base models on a large corpus of over 20 million medical articles, producing our biomedical language model, BioELECTRA. We finetune BioELECTRA-Small and BioELECTRA-Base on biomedical QA using general-domain and biomedical-domain QA datasets. Finally, we evaluate our solution on gold-standard biomedical questions.

Results – Our BioELECTRA-Base binary classification model achieves an average Macro F_1 score of 0.722 on yes/no questions. For factoid and list questions, our BioELECTRA-Base extractive model achieves an average MRR score of 0.364 and an average F_1 score of 0.346, respectively. BioELECTRA outperforms ELECTRA by 12.3%, 72.5% and 73.0% for yes/no, factoid and list questions.

Conclusions – We develop a biomedical language model despite resource constraints by utilising the highly-efficient ELECTRA pretraining objective. We demonstrate that pretraining ELECTRA on biomedical corpora significantly improves biomedical QA performance. Finally, we show that increasing model size has a lesser effect on performance than additional finetuning on general-domain QA datasets.

***Keywords* —** Biomedical Question Answering, Transfer Learning, Transformers, Natural Language Processing, Machine Learning, Deep Learning

I INTRODUCTION

Due to the current COVID-19 pandemic, there has been a rapid increase in the rate of publication of medical research in online journals [7]. While this research is high-quality and valuable to society, distributing this information as quickly as it is published is challenging. Advancements in biomedical question answering research could vastly improve access to such reliable health information; allowing healthcare professionals to query published literature quickly and widening access to help tackle misinformation in the rest of society. Providing highly-accurate answers to medical questions presents a significant challenge, as it is ethically-necessary to protect consumers from receiving misleading or harmful information.

In recent years, transfer-learning approaches utilising complex language models have dominated question answering (QA) research. Over time, research has revealed a strong correlation between greater computing resources and improved performance; hence, it is becoming infeasible to train recently-released models with limited resources. Researchers will often pretrain popular models on general-domain corpora and release the pretrained weights to allow others to use them 'off-the-shelf'. However, due to the significant difference in biomedical and general-domain vocabulary, general-domain language models cannot accurately interpret biomedical text. Hence, numerous studies have deemed it necessary to pretrain models specifically on biomedical corpora in order to achieve competitive results in biomedical QA [10, 13, 29]. This highlights the main challenge of biomedical QA and the scope for utilising an efficient pretraining approach.

A Problem Background

In the medical field, healthcare practitioners are required to keep up with ever-evolving medical research. Health information specialists currently play an important role in providing up-to-date information to healthcare practitioners [2]. However, this role requires a lot of expertise and training; hence, there is motivation to automate this process. Improvements to biomedical question answering systems could enable medical practitioners to gather the most up-to-date information, relating to a query, without the need for an intermediary. As humans, we develop vast knowledge bases which enable us to identify and respond to different question types; however, it is difficult to simulate this with an automated question answering model. For instance, some questions require a yes/no answer, whereas others require a summary, a short factoid answer, or multiple short answers.

Like in many NLP problems, transfer-learning techniques are the current state-of-the-art in question answering. This is where a neural network is first pretrained on a data-rich problem to produce a language model with an intricate knowledge of language. Then, the model is fine-tuned on a downstream task, such as question answering, to tweak its parameters to fit the specific problem. In earlier approaches, language-specific knowledge had to be explicitly incorporated into question answering models using pre-computed word embeddings, such as those of Word2Vec [15] and GloVe [19]. More advanced language models, such as ELMo [20], could disambiguate polysemic words by producing deep contextualised word-representations on-the-fly. However, a major advantage of transfer learning is that a pretrained language model can be repurposed to perform question answering with very few architectural modifications, rather than having to train a separate model on the word embeddings produced by previous techniques.

In this project, we use a transfer-learning approach to develop a set of biomedical question answering models. As many recent models rely upon vast quantities of data coupled with extensive training, we use the ELECTRA model [4] which maximises learning in resource-constrained settings. ELECTRA achieves the performance of other state-of-the-art language models, such as GPT [21] and RoBERTa [14], with one quarter of their computing resources. Hence, it is a model well-suited to problems with resource constraints. We adapt ELECTRA for the biomedical domain by pretraining on biomedical corpora and finetuning on biomedical question answering.

B Research Questions and Objectives

The **research question** guiding this project is – *Can ELECTRA achieve state-of-the-art performance in biomedical question answering when pretrained on biomedical corpora?*

The objectives below are designed to address this research question, and are divided into three categories in accordance with their priority level and difficulty.

The **minimum** objectives of this project are to collate and pre-process a vast pretraining corpus of medical text with which to pretrain the small and base ELECTRA models. In addition, we aim to finetune and evaluate our pretrained models on a biomedical question answering dataset. In order to fulfil these objectives, we first collect a corpus 20 million medical articles from an online repository of medical text. We use this corpus to pretrain ELECTRA-Small and ELECTRA-Base using *replaced token detection*, producing the BioELECTRA-Small and BioELECTRA-Base models. Finally, we finetune and evaluate on the BioASQ (biomedical question answering) dataset, where we compare the performance against the state-of-the-art.

The **intermediate** objectives of this project include training a biomedical tokeniser in order to create efficient representations of medical text. We achieve this objective by training a custom WordPiece tokeniser on our biomedical pretraining corpus. Furthermore, we compare its ability to represent medical text concisely in contrast to an equivalent tokeniser trained on general-domain text, discovering that our custom tokeniser represents abstracts from our pretraining corpus using 16.7% fewer tokens than the general-domain tokeniser. Additionally, we aim to explore whether finetuning BioELECTRA on general-domain QA prior to biomedical QA improves biomedical question answering performance. In order to fulfil this objective, we utilise two additional datasets in the finetuning phase - SQuAD [24] and BoolQ [3].

The **advanced** objectives of this project include comparing ELECTRA, instantiated with the pretrained weights from Clark et al. (2020) [4], to BioELECTRA on biomedical question answering. We also aim to submit our solution to the BioASQ challenge (2021). In order to achieve these aims, we assess the performance improvement attained by utilising BioELECTRA over ELECTRA on the BioASQ dataset, observing an improvement of 12.3%, 72.5% and 73.0% for yes/no, factoid and list questions. We submit both the small and base implementations of BioELECTRA to the 9th edition of the BioASQ challenge in order to allow the wider research community to compare BioELECTRA to state-of-the-art solutions. Our final advanced objective is to develop a user-friendly interface to allow users to put customised queries to our solution. We achieve this by creating a web-interface within which we load our best-performing models.

II RELATED WORK

Significant research has been published in the realm of general-domain question answering, whereby questions span an unbounded collection of topics. This project is centered around biomedical question answering, a closed-domain problem, although many general-domain techniques can be adapted for a smaller collection of related topics. Many of the state-of-the-art solutions in biomedical question answering originate from task B phase B of the annual BioASQ challenge, which garnered 94 submissions in 2020 [16].

A Background on Transformers

Emerging methodologies in the wider field of natural language processing, specifically the application of transfer learning, have changed the landscape of question answering approaches. As the Transformer model [26] is a fundamental component of recent models harnessed for transfer learning, such as BERT [5] and ELECTRA [4], we provide a brief description of its utility and an illustration of its architecture in figure 1.

Initially designed for sequence-to-sequence problems, the transformer demonstrates that a simple architecture utilising an attention mechanism can contend with highly performant recurrent models, such as LSTMs. The attention mechanism allows the transformer to attend to states at earlier points in the input, drawing knowledge from the most relevant parts of a sequence.

Transformers contain several stacked layers of encoders (encoder block) and decoders (decoder block). Encoders produce an intermediate representation from an input sequence, incorporating information supplied by the attention mechanism to highlight relevant tokens. The final encoder passes this data to the decoder block, whereby the final decoder layer produces an output sequence by attending to the outputs of previous decoders. In the remainder of this section, we present state-of-the-art question answering systems, many of which are Transformer-based language models.

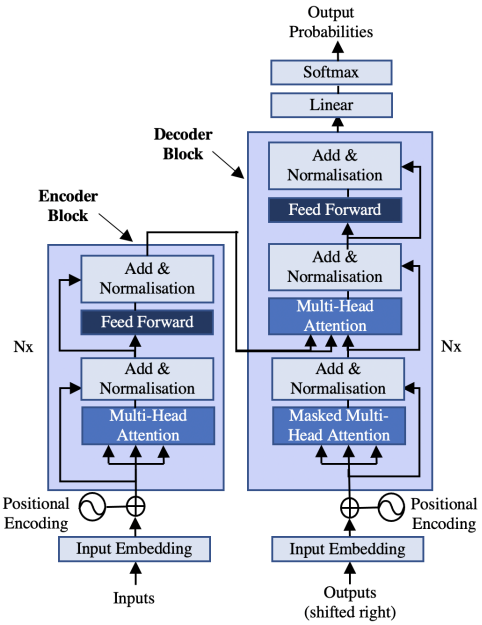


Figure 1: Transformer Architecture

B Extractive Question Answering

The task of extractive question answering is formulated as follows: given a question and a context paragraph, the start and end positions of an answer span are predicted from the context paragraph (figure 2). Consisting of over 100,000 questions, SQuAD [24] is a benchmark extractive question answering dataset covering a vast collection of general-domain topics. Since its release in 2016, many researchers have chosen to tackle general-domain and biomedical-domain question answering as an extractive problem.

Question:
What nerve is involved in carpal tunnel syndrome?

Context:
Carpal tunnel syndrome (CTS) is a focal compressive neuropathy of the median nerve at the level of the wrist.

Figure 2: Example factoid question

Domain Adaptation — In recent years, deep-learning approaches have become the state-of-the-art in question answering (QA), enabled by the release of large-scale datasets such as SQuAD. Due to the increased cost of collating specialised datasets, biomedical-domain datasets tend to be smaller as they are curated by a team of medical experts [16]. Consequently, many approaches focus on applying knowledge from trained general-domain systems to closed-domain QA.

Wiese et al. (2017) [29] utilises a domain-adaptation technique to transfer knowledge from FastQA [28], a Bidirectional Recurrent Neural Network, to the biomedical domain. As one of the first deep neural approaches to biomedical question answering, they train FastQA on static contextualised word-embeddings, including GloVe [19] and Biomedical Word2Vec [18] embeddings. They pretrain FastQA on the SQuAD dataset and finetune on the BioASQ 5B training set, adjusting FastQA's parameters to biomedical QA. Wiese et al. (2017) achieved strong performance on factoid and list questions in the 2017 BioASQ challenge, obtaining an MRR score of 0.405 and an F_1 score of 0.329 on factoid and list questions (table 1).

In *Lee et al. (2019)* [13], they develop a specialised biomedical language model, known as BioBERT, by pretraining the popular BERT model [5] on biomedical corpora. Due to the difference in word distributions of biomedical and general-domain text, they hypothesise that language models must be pretrained on biomedical corpora in order to be effective on downstream biomedical text-mining tasks. As Wiese et al. [25] demonstrates that pretraining on SQuAD improves biomedical question answering performance, they finetune BioBERT on both SQuAD and BioASQ. BioBERT exceeded the performance of its competitors in the biomedical question answering phase of the 7th BioASQ challenge [31], averaging a Macro F_1 score of 0.717, an MRR score of 0.512 and an F_1 score of 0.406 across yes/no, factoid and list questions, respectively. However, achieving this result required more than 3 weeks of continuous training across 8 GPUs - highlighting the difficulty of pretraining large transformer-based language models.

Kommaraju et al. (2020) outlines another transfer-learning technique, focusing on transferring knowledge from general-domain question answering datasets to biomedical question answering. They train a model, based on BioBERT, using a novel pretraining approach in which entities are identified in biomedical text and intentionally corrupted with noise. The model then attempts to denoise the text by locating the corrupted segment; this helps the model learn to distinguish between regular and distorted biomedical text in a process similar to that of Replaced Token Detection in ELECTRA [4] (described in depth in section C.1). In a similar fashion to Lee et al. (2019), they finetune their model on SQuAD and BioASQ, exceeding the performance of BioBERT on yes/no and list questions in the 2020 BioASQ challenge (table 1).

Sentence Similarity — In Laskar et al. (2020), [12], they approach extractive question answering as a sentence similarity problem, hypothesising that sentences in the context paragraph that are most similar to the question are most likely to contain the answer. They utilised contextualised embeddings from ELMo, BERT, and RoBERTA [14] in feature-based and finetuning methods of extracting answers from a context paragraph. They conducted experiments on six general-domain question answering datasets, finding that finetuning approaches outperformed feature-based approaches across the board. Most notably, they achieved state-of-the-art performance on all six datasets using the pretrained RoBERTa model. Another sentence similarity approach [6] was trialled explicitly in biomedical question answering and submitted to the BioASQ Challenge in 2020, utilising the pretrained language model, BioBERT [13] and a logistic regression model.

Table 1: Comparison of Submissions to Task B Phase B of BioASQ

Model	**Approach**	**Task**	**Yes/No** *(Macro F_1)*	**Factoid** *(MRR)*	**List** *(Avg. F_1)*
Wiese et al. (2017)	BiRNN with biomedical word-embeddings	5B	N/A	0.405	0.329
Lee et al. (2019)	Biomedical transfer learning	7B	0.717	0.512	0.406
Han and Tsai. (2020)	Cosine similarity of BioBERT embeddings	8B	0.721	0.496	0.341
Kommaraju et al. (2020)	Unsupervised representation learning	8B	0.760	0.439	0.431

Table 1 compares the scores obtained by each approach in previous BioASQ challenges. The official evaluation metrics for the BioASQ challenge are Macro F_1, MRR and average F_1 score.

C Generative Question Answering

Given that extractive approaches to question answering rely upon the provision of a supplementary context document, unanswerable questions arise when a context paragraph does not contain the answer. This is one of the main limitations of extractive question answering models; however, generative language models can produce original samples of text given an initial prompt or question, allowing us to generate original answers without the need for a context document.

Introduced in *Radford et al. (2019)*, GPT-2 [22] is a flexible general-domain generative language model. The aim of this model is to perform down-stream tasks without any architectural modifications by conducting unsupervised training on a vast corpus (40GB) of web-text. The intuition is that task-specific data will become embedded in this diverse corpus e.g. questions followed by answers can incorporate question answering knowledge without explicit training on this task. However, when evaluating on the Natural Questions dataset [11], they found that question answering performance is weak as only 4.1% of questions are answered correctly.

In *Raffel et al. (2019)* [23], they develop a generative language model, known as the Text-To-Text Transformer model (T5), achieving a significant boost to question answering performance. They pretrain T5 using a highly similar approach to that of Radford et al. (2019), however, they conduct *supervised* finetuning on question answering. When finetuned on SQuAD, T5 achieved an F1 score of 88.81, and an exact match score of 90.88, highlighting the scope for generative models in future question answering research. Applying a similar principle to BioBERT, if a generative language model were to be pretrained on a large corpus of biomedical text and explicitly finetuned on biomedical question answering, this could produce a domain-specific question answering model which does not rely on a context paragraph.

III SOLUTION

This project addresses the problem of question answering for multiple question types, aiming to provide an accurate response to a medical question given a supplementary context paragraph. We develop a trained neural language model, named BioELECTRA, to tackle this problem; hence, this is the focus of the following section. Two main phases of training are conducted - pretraining on biomedical corpora and finetuning on biomedical question answering. In general, the aim of the pretraining stage is to develop a language model capable of understanding text, before finetuning the model on a specific downstream task (i.e. question answering). An overview of the pretraining and finetuning procedures is provided in figure 3 and figure 4.

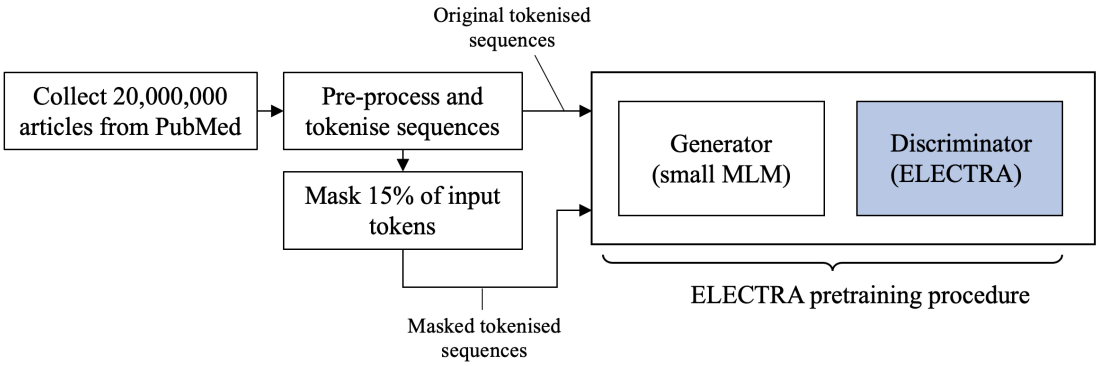


Figure 3: Overview of the *pretraining* stage

Prior to pretraining, we collect and pre-process a random selection of 20 million articles from PubMed[1], an online repository of over 30 million citations and abstracts from medical publications[2]. We conduct pretraining using a small masked language model (MLM) and a discriminator (discussed in depth in section C.1) in order to produce our biomedical language model. In accordance with the ELECTRA pretraining objective, 15% of input tokens are masked and provided as input to the generator, alongside the original tokenised sequences.

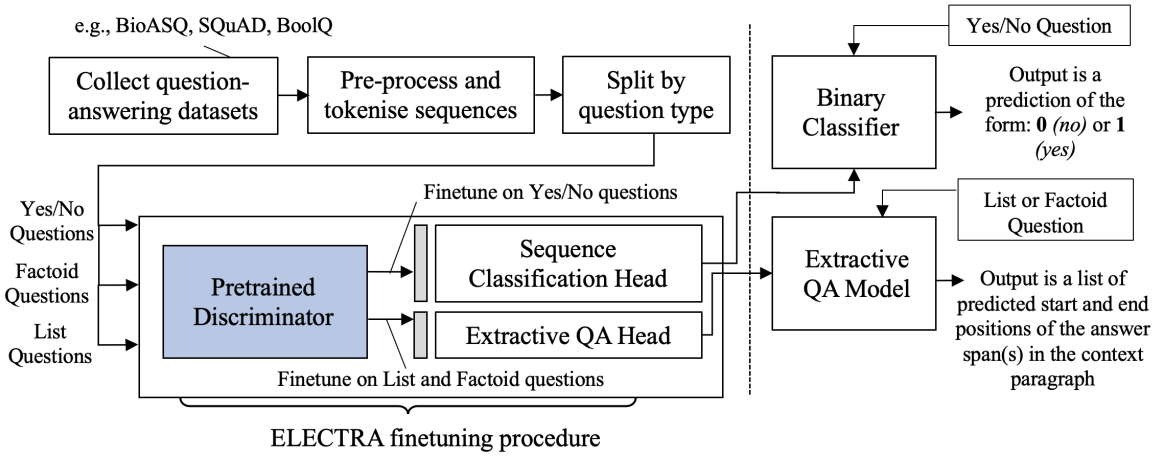


Figure 4: Overview of the *finetuning* stage

During the finetuning stage, we build a binary classifier for yes/no questions and an extractive question answering model for factoid and list questions. We initialise our discriminator with the pretrained weights learned during the pretraining stage, allowing our model to carry forward its learned knowledge of biomedical language. We enable our model to make answer predictions, given a question and context paragraph, by making small architectural modifications to the discriminator (explained in depth in section D).

A Implementation tools

The main contribution of this project is developed in Python, supplemented with external libraries to hasten the development of our deep neural language model. We use the Transformers library from Huggingface [30] as it contains a large collection of transformer-based models, including the generator and discriminator models which form the core components of our pretraining setup. Furthermore, it contains pretrained weights associated with other question answering models to allow for comparison. We utilise PyTorch [17], a machine-learning framework in Python, in order to implement pretraining and finetuning processes for our model. Additionally, a web-interface for interacting with our solution is developed using Flask, JavaScript, HTML and CSS.

B Tokenisation

Due to the usage of high-quality medical datasets, we apply a simple normalisation technique (conversion to lower-case) prior to tokenisation. WordPiece tokenisation [25] is used to split sentences into sequences of tokens (or word pieces) and the token ids are passed as input to our models. Due to memory constraints, we are limited in the maximum sequence length we can pass

[1]Data obtained from pubmed.ncbi.nlm.nih.gov

[2]Statistic obtained from pubmed.ncbi.nlm.nih.gov/about

to our models; for instance, the ELECTRA-Small model takes sequences of length 128 tokens. Given this limitation, we need to be able to represent text as concisely as possible in order to maximise the information-density of our input sequences.

With WordPiece tokenisation, a word is represented either as a whole word token if it appears in the vocabulary, or as multiple composite sub-tokens. We train a state-of-the-art WordPiece tokeniser on our pretraining corpus of 20 million medical articles, allowing our tokeniser to build a specialised medical vocabulary. This means that frequently-occurring medical words are more likely to appear as whole word tokens (e.g. coronary), rather than multiple sub-tokens (e.g. *cor ##on ##ary*), than if we had used an off-the-shelf general-domain tokeniser. Hence, we can provide our model with richer features given the input length restrictions.

Table 2: Avg. sequence length by tokeniser

Tokeniser	**Average Sequence Length**
General-domain	225.04
Biomedical-domain	192.78

We conduct an experiment to compare the average length of tokenised abstracts (from our pretraining corpus) produced by the general-domain tokeniser and the biomedical-domain tokeniser to investigate this idea further. Table 2 shows that the biomedical-domain tokeniser is able to represent input sequences using 16.7% fewer tokens; hence, this tokeniser is used in the final implementation of our solution.

C Pretraining

As our solution utilises transfer learning, we first conduct pretraining on a corpus of 20 million medical articles from PubMed - an online repository of citations and abstracts from medical journals. As we utilise the ELECTRA model, two neural networks are required during pretraining - a generator and a discriminator. The discriminator is trained as a biomedical language model using ELECTRA's unique pretraining objective. We provide an outline of the pretraining objective which makes ELECTRA a good candidate for tackling our problem.

C.1 Replaced Token Detection

The ELECTRA model [4] utilises an efficient pretraining objective, known as *Replaced Token Detection*. In Replaced Token Detection (*figure 5*), a generator and a discriminator are trained jointly, although only the pretrained discriminator is utilised in finetuning.

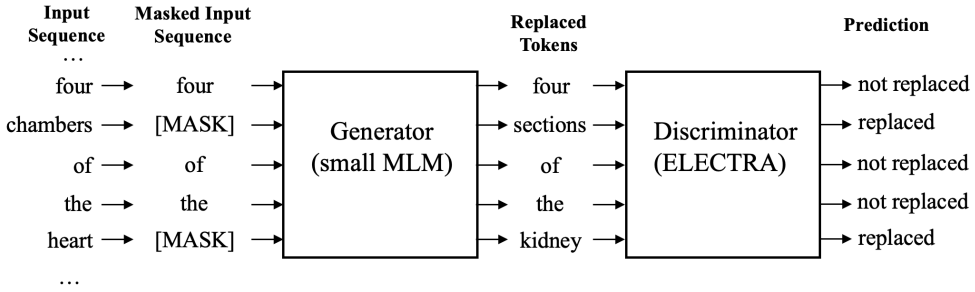


Figure 5: Replaced Token Detection on medical text

Given a sequence of tokens, whereby a fixed proportion (typically 15%) are replaced with the [MASK] token, it is the task of the generator to replace the masked tokens with realistic alternatives. The discriminator uses a sigmoid output layer to predict, for each token, whether it has been replaced by the generator, or whether it belongs to the original un-masked sequence. Over time, the discriminator learns which tokens are likely to make sense in the sequence and which are out of place. In language, words have meaning due to context in which they appear; hence, pretraining the discriminator in this way produces an intelligent biomedical language model. The motivation for training the generator alongside the discriminator is to enable the generator to provide challenging negative samples to the discriminator, rather than sampling randomly from the entire vocabulary.

In comparison to the Masked Language Modelling (MLM) mechanism used in BERT [5], replaced token detection is more efficient and leads to better performance, allowing ELECTRA to compete with other transformer-based language models requiring far more compute resources. This is due to the loss being defined over all input tokens; hence, the model can learn from all input tokens, rather than just the masked 15% like in BERT. In addition, replaced token detection moves away from passing the [MASK] token to the discriminator, avoiding the discrepancy in BERT of presenting [MASK] tokens during pretraining but not finetuning.

C.2 Pretraining Settings

The model settings used to produce the small and base models of BioELECTRA are given in table 3, where they differ from those used in the original implementation of ELECTRA [4]. As we are limited to a single GPU with 24GB of virtual-ram, we are constrained to using the ELECTRA-Small and ELECTRA-Base models with a few tweaks to reduce memory usage.

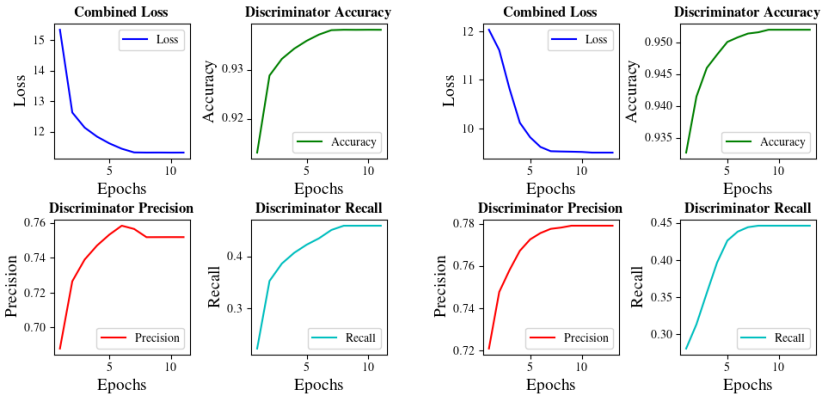
For the base model, we reduce the batch-size from 256 as recommended in the original paper to 32 due to memory constraints. This enables us to use the maximum input sequence length of 256 tokens. Longer input sequences are preferable for question answering as this enables additional context to be provided to the model, improving answer prediction. Early experimentation with the batch size at 32 and the recommended learning rate of $2e-4$ led to the discriminator overfitting on the *not replaced* class, hence we reduce the learning rate to $2.5e-5$.

Table 3: Pretraining Settings

	Model Size	
Settings	Small	Base
Max Epochs	15	20
Batch Size	128	32
Learning Rate	$5e-4$	$2.5e-5$
Max Length	128	256

Figure 6 shows the pretraining curves for the discriminator, measured at the end of each epoch. The precision, recall and accuracy metrics measure the discriminator's ability to correctly predict which tokens have been replaced by the generator. The combined loss represents the sum of the weighted loss of the generator and discriminator. We focus on improvements to the discriminator, rather than the generator, as this forms the initial model for the finetuning phase of the solution.

We pretrain our models until the maximum number of epochs is reached or until their performance plateaus. For BioELECTRA-Small (figure 6a), pretraining provides no additional benefit after epoch 8, when each metric plateaus. For the BioELECTRA-Base (figure 6b), recall shows a steep increase within the first 6 training epochs before plateauing, while accuracy and precision increase more steadily over 8 epochs.



(a) BioELECTRA-Small Model (b) BioELECTRA-Base Model

Figure 6: Pretraining curves

D Finetuning

Prior to finetuning, we make small architectural modifications to BioELECTRA (as illustrated in figures 8 and 9) to enable our model to make answer predictions. We conduct finetuning on our modified BioELECTRA-Small and BioELECTRA-Base models to tune their parameters to question answering. We utilise the gold-standard training and testing data from the 2020 edition of the annual BioASQ challenge [16]. The training and testing datasets contain yes/no, factoid and list questions - each constructed by a team of medical experts. Table 4 highlights one of the major challenges in biomedical question answering, as there are very few questions across all three question types, with significantly fewer list questions (comprising 25.28% of the dataset).

Yes/No Questions — Given a question and a collection of context snippets, the model predicts a label of *yes* or *no*; hence, we approach yes/no questions as a **binary classification** problem.

Factoid and List Questions — Given a question and a collection of text snippets, the model predicts the start and end positions of an answer span in each snippet. For factoid questions, a single answer span is predicted, whereas multiple answers are predicted for list questions. We approach factoid and list questions as an **extractive question answering** problem.

Table 4: Composition of the BioASQ training dataset

Question Type	Num Samples	% Samples	Avg. Question Length	Avg. Context Length
List	719	25.28%	63.49	264.93
Factoid	1092	38.40%	61.02	238.84
Yes/No	1033	36.32%	60.98	200.09

Prior to finetuning, we convert the question and context snippets to lowercase before performing WordPiece tokenisation. We concatenate the tokenised question and context (as shown in figure 7) with a separator token ([SEP]) to indicate where the sequence is partitioned. As the BioASQ dataset provides several context snippets per question, we produce multiple question-context pairs. Consequently, several answers are produced per question. In section D.1 and D.2, we outline the post-processing steps taken for each question type to aggregate answers accordingly.

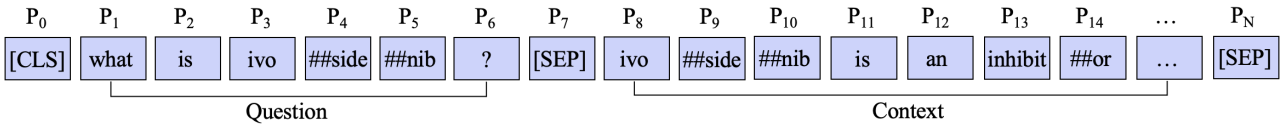


Figure 7: Example Tokenised Sequence

D.1 Binary Classification

In order to modify BioELECTRA for classifying yes/no questions, we append a binary classification head (figure 8) to transform the hidden vectors from BioELECTRA into class probabilities. For yes/no questions, we classify the BioELECTRA representation for the [CLS] token. Dropout is used as a regularisation technique to reduce the likelihood of the model over-fitting to the training data, and the final softmax function produces probabilities of the question belonging to the *yes* class or the *no* class. As the model produces several predictions per question, we take the majority label as the final prediction.

We finetune our binary classification model on yes/no questions from the BioASQ dataset. In addition, we assess whether prior finetuning on general-domain yes/no questions from the BoolQ dataset [3] can improve biomedical question answering performance on the BioASQ dataset.

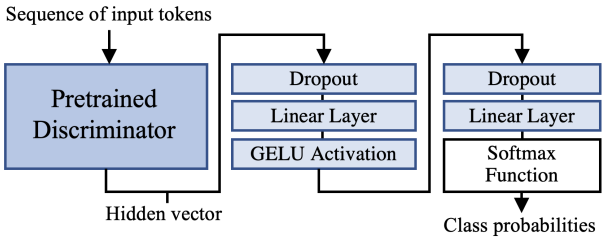


Figure 8: Binary Classification Head

For yes/no questions in the BioASQ dataset, the data is heavily imbalanced, with 77.35% positive instances and 22.65% negative instances. To prevent over-fitting on the majority (yes) class, we consider under-sampling the dataset where samples from the majority class are ignored. However, as the dataset is small to begin with, this technique would produce an even smaller dataset and exclude many samples that the model could learn from. We also consider over-sampling the dataset by duplicating entries in the minority class; however, this can lead to over-fitting as the model can devise simple classification rules to cover these replicated examples [27].

Instead of re-sampling the dataset, we implement a weighted Binary Cross-Entropy loss function which punishes our model for incorrect predictions on the minority class. For each class, i, we compute a weight W_i using the formula in equation 1, where $C = \{0, 1\}$ is the set of classes and S_i is the number of samples in class i. Applying weighting using this technique ensures that the class with the fewest samples has the greatest weight.

$$W_i = \frac{max(S_i \forall i \in C)}{S_i} \tag{1}$$

D.2 Extractive Question Answering

We modify BioELECTRA for extractive question answering by adding a linear layer which transforms the hidden vector produced by BioELECTRA to start and end logits. The argmax function is applied to the start and end logits to find the most likely positions of the start and end of the answer span. We use the Cross-Entropy loss function as extractive question answering can be framed as a multi-class classification problem, where the probability of each token being the start (or end) position is returned. We compute the loss for start position predictions, $Loss_{start}$, and end position predictions, $Loss_{end}$, then take the mean of $Loss_{start}$ and $Loss_{end}$.

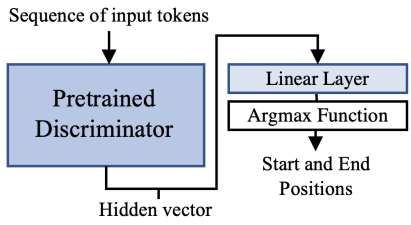


Figure 9: Extractive Question Answering Head

For factoid questions, each training example consists of a question, context snippet and a *single answer*. As list questions have multiple correct answers, we apply an additional pre-processing step to replicate this format by duplicating question-context pairs and matching them to each correct answer. This allows us to train a single extractive QA model and increases the size of the training set as we train on factoid and list questions together. We exclude 28% of training examples from the BioASQ dataset where the exact answer cannot be retrieved from the context. In addition to training our extractive model on BioASQ, we assess whether prior finetuning on general-domain factoid questions from the SQuAD dataset [24] can improve biomedical question answering performance.

We merge the list of predictions and their probabilities for each question-context pair. For factoid questions, we take the most probable answer as the final prediction. However, as list questions require multiple answers, we implement a probability thresholding approach. If the most probable answer, x, has probability P_x, then a prediction, i, is included in the final prediction list if $P_i > t \cdot P_x$, where t is the threshold. In the BioASQ challenge, penalties are applied where too many or too few list predictions are returned by the model; hence, we experiment with different values for the threshold. We find that values less than 0.85 result in a reduction in average precision as too few values are returned, whereas values higher than 0.85 result in a lower average recall as too many values are returned. Hence, we set the hyperparameter t to 0.85.

D.3 Finetuning Settings

Table 5 shows the settings used to finetune BioELECTRA on each dataset. With transfer learning, finetuning typically requires fewer training steps than pretraining before optimal performance is achieved on the validation set. In order to apply the optimal number of finetuning steps for our models, we utilise check-pointing to save the weights of the model which minimises validation loss during training. We train for a maximum of 12 epochs, as early experimentation revealed that validation loss plateaus long before this milestone is reached.

Table 5: Finetuning Settings

	Extractive QA		**Binary QA**	
Settings	SQuAD	BioASQ	BoolQ	BioASQ
Max Epochs	3	2	2	2
Learning Rate	$3e-4$	$2e-4$	$1e-4$	$2e-4$

For the majority of the finetuning settings, we maintain consistency by utilising the settings outlined in the ELECTRA paper [4]. For instance, we implement layer-wise learning rate decay for the finetuning stage, such that layers closer to the input have a lower learning rate than those further away. In addition, we employ the the Adam optimiser [8] with default parameters of $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e-8$. We discover that modifying the learning rate for finetuning on each dataset has a positive impact on our results. For example, a learning rate of $1e-4$ is found to be optimal in finetuning on the BoolQ dataset - higher learning rates prevent our model from learning. Similarly, we discover that a learning rate of $2e-4$ is optimal for finetuning on the BioASQ dataset. The default learning rate of $3e-4$ used in the ELECTRA paper [4] is found to be optimal for SQuAD.

IV RESULTS

A *Performance Metrics*

The following metrics are used to evaluate the biomedical question answering performance of our finetuned models and they are also the official evaluation metrics of the BioASQ challenge.

$$Precision = \frac{TP}{TP\ +\ FP} \quad (2) \qquad Recall = \frac{TP}{TP\ +\ FN} \quad (3)$$

$$F_1\ Score = \frac{2\ \times\ Precision\ \times\ Recall}{Precision\ +\ Recall} \quad (4) \qquad MRR = \frac{1}{|N|} \cdot \sum_{i=1}^{|N|} \frac{1}{r(i)} \quad (5)$$

Precision (*equation 2*) refers to the proportion of *positive answers* (i.e. yes) which are truly positive (for yes/no questions), or the proportion of predicted answers which are also ground truth answers (for list questions). Similarly, for yes/no questions, recall (*equation 3*) is the proportion of *truly positive* answers which are ultimately classified as such, and for list questions, recall is the proportion of ground truth answers that feature in the list of predictions. In addition, we utilise the harmonic mean of precision and recall, known as the F_1 score *(equation 4)*.

For factoid questions, a list of candidate answers is returned, ranked from most to least probable. We use two variations of the standard accuracy metric - *strict accuracy* and *lenient accuracy*. Strict accuracy considers an answer correct if it is in position 1 (ranked most likely), while lenient accuracy considers an answer correct if it places anywhere in the list of candidate answers. Mean Reciprocal Rank *(equation 5)* is also used to evaluate our dataset of N factoid questions by rewarding models which rank the correct answer more highly in their list of predictions. If the correct answer is in position k, r(i) = k, otherwise, $r(i) \rightarrow +\infty$; hence, $\frac{1}{r(i)} = 0$.

A.1 Binary Classification Results

Table 6 shows the results of our binary classification models on yes/no questions from each of the five data batches provided in the 2020 BioASQ challenge. Models A and B are finetuned from BioELECTRA-Small and models C and D are finetuned from BioELECTRA-Base. Furthermore, models A and C are finetuned only on BioASQ, while models B and D are additionally finetuned on BoolQ [3] - a dataset of general-domain yes/no questions.

Table 6: Results for Yes/No Questions

	Batch 1		**Batch 2**		**Batch 3**		**Batch 4**		**Batch 5**	
Small Models	A	B	A	B	A	B	A	B	A	B
Accuracy	0.720	0.760	0.722	0.750	0.710	0.750	0.654	0.692	0.706	0.676
F_1 Yes	0.811	0.842	0.821	0.847	0.791	0.842	0.757	0.765	0.773	0.718
F_1 No	0.462	0.500	0.375	0.387	0.527	0.720	0.400	0.556	0.584	0.621
Macro F_1	0.637	<u>0.671</u>	0.598	0.617	0.659	0.765	0.579	0.661	0.678	0.669
Base Models	C	D	C	D	C	D	C	D	C	D
Accuracy	0.760	0.768	0.806	0.722	0.774	0.806	0.731	0.746	0.706	0.765
F_1 Yes	0.842	0.840	0.857	0.815	0.810	0.842	0.800	0.816	0.750	0.810
F_1 No	0.500	0.528	0.400	0.444	0.700	0.750	0.589	0.688	0.643	0.692
Macro F_1	<u>0.671</u>	**0.684**	<u>0.621</u>	**0.629**	<u>0.778</u>	**0.796**	<u>0.695</u>	**0.752**	<u>0.697</u>	**0.751**
Benchmark*	0.603 / 0.866		0.700 / 0.926		0.622 / 0.903		0.685 / 0.845		0.743 / 0.853	

Benchmark refers to the median/best Macro F_1 scores of the yes/no models submitted to the 2020 BioASQ challenge, as this is the official evaluation metric for yes/no questions. The highest scores are shown in **bold** and the second-highest scores are underlined.

Finetuning on BoolQ prior to BioASQ increases the Macro F_1 score by an average of 7.36% and 4.51% for our small and base models, respectively. This shows that finetuning on general-domain question answering is a worthwhile finetuning step for both BioELECTRA-Small and BioELECTRA-Base, with BioELECTRA-Small benefitting the most. BioELECTRA-Base, finetuned only on BioASQ, obtained an average Macro F_1 score of 0.691 - exceeding the performance of both models finetuned from BioELECTRA-Small. BioELECTRA-Base, finetuned on BoolQ and BioASQ is our best performing binary classification model, achieving an average Macro F_1 score of 0.722 - an average increase of 7.72% from the average Macro F_1 score of median submission to the 2020 BioASQ challenge.

A.2 Extractive Question Answering Results

Tables 7 and 8 show the results of our extractive question-answering models on factoid and list questions from each of the five data batches provided in the 2020 BioASQ challenge. Models A and B are finetuned from BioELECTRA-Small, whilst models C and D are finetuned from BioELECTRA-Base. Models A and C are finetuned only on BioASQ, while models B and D are additionally finetuned on SQuAD - a dataset of general-domain factoid questions.

Table 7: Results for Factoid Questions

	Batch 1		**Batch 2**		**Batch 3**		**Batch 4**		**Batch 5**	
Small Models	A	B	A	B	A	B	A	B	A	B
Strict Acc	0.250	0.281	0.080	0.240	0.143	0.179	0.235	0.382	0.250	0.438
Lenient Acc	0.406	0.406	0.120	0.280	0.250	0.214	0.412	0.471	0.375	0.500
MRR	0.304	<u>0.336</u>	0.100	<u>0.260</u>	0.175	0.196	0.294	<u>0.413</u>	0.292	**0.464**
Base Models	C	D	C	D	C	D	C	D	C	D
Strict Acc	0.219	0.344	0.120	0.280	0.179	0.214	0.265	0.382	0.312	0.406
Lenient Acc	0.438	0.469	0.240	0.320	0.321	0.321	0.382	0.500	0.500	0.500
MRR	0.293	**0.397**	0.180	**0.288**	<u>0.233</u>	**0.253**	0.311	**0.431**	0.372	<u>0.453</u>
Benchmark*	0.316 / 0.469		0.233 / 0.353		0.281 / 0.397		0.521 / 0.628		0.538 / 0.635	

Benchmark refers to the median/best MRR scores of factoid models submitted to the 2020 BioASQ challenge, as this is the official evaluation metric for factoid questions.

In comparing our small and base models; we observe an average MRR increase of 16.6%. However, when we focus on models finetuned on both SQuAD and BioASQ, we observe a much smaller increase of 8.3% between model B (BioELECTRA-Small) and model D (BioELECTRA-Base). In fact, model B routinely achieves second place above our BioELECTRA-Base model finetuned only on BioASQ (model C), and even outperforms model D on batch 5. This suggests that additional finetuning on general-domain question answering is more beneficial to biomedical factoid question answering performance than increasing the number of trainable parameters in BioELECTRA. Overall, our BioELECTRA-Base model finetuned on both SQuAD and BioASQ

achieves the highest MRR scores on 4 out of 5 batches of data. We achieve an average of 0.260 and 0.371 for strict and lenient accuracy, suggesting that 70% of correctly predicted answers are ranked in position 1 (most-probable). While we do not outperform the state-of-the-art, we exceed the median MRR score of BioASQ 2020 submissions by 25.6% and 23.6% on batches 1 and 2.

Table 8: Results for List Questions

	Batch 1		**Batch 2**		**Batch 3**		**Batch 4**		**Batch 5**	
Small Models	A	B	A	B	A	B	A	B	A	B
Avg Precision	0.212	0.214	0.257	0.269	0.319	0.450	0.231	0.266	0.333	0.389
Avg Recall	0.312	0.332	0.498	0.519	0.414	0.537	0.386	0.387	0.440	0.473
Avg F_1 Score	0.249	<u>0.255</u>	0.314	<u>0.332</u>	0.312	<u>0.426</u>	0.266	<u>0.289</u>	0.359	**0.396**
Base Models	C	D	C	D	C	D	C	D	C	D
Avg Precision	0.204	0.221	0.257	0.276	0.402	0.447	0.253	0.292	0.347	0.354
Avg Recall	0.297	0.329	0.495	0.531	0.518	0.532	0.407	0.445	0.448	0.425
Avg F_1 Score	0.238	**0.260**	0.318	**0.342**	0.397	**0.432**	0.285	**0.327**	0.363	<u>0.368</u>
Benchmark*	0.315 / 0.432		0.376 / 0.474		0.349 / 0.502		0.336 / 0.457		0.365 / 0.525	

**Benchmark* refers to the median/best average F_1 Score of list models submitted to the 2020 BioASQ challenge, as this is the official evaluation metric for list questions.

On average, our BioELECTRA-Base models (C and D) achieve an average F_1 score that is 4.13% higher than our BioELECTRA-Small models (A and B). Similar to factoid questions, we find that model B outperforms model C on every batch; hence, finetuning on general-domain question answering results in a greater performance boost to BioELECTRA-Small than upgrading the model size to BioELECTRA-Base. While this is an important feature to note, the best-performing model utilises a combination of both upgrades - as BioELECTRA-Base, finetuned on SQuAD and BioASQ, achieves the highest average F_1 scores across 4 out of 5 batches.

Across all of our experiments, we find that recall is consistently higher than precision, suggesting that the lists of predictions produced by our models contain many of the expected answers; however, they also include a lot of noise. To combat this, we experiment with increasing our probability threshold above 0.85 to filter out less confident answers. However, this results in a small increase in precision and a large decrease in recall (and therefore MRR score). Our solution exceeds the median average F_1 score of models submitted to the 2020 BioASQ challenge on batches 3 and 5.

V EVALUATION

The strengths and limitations of our solution are evaluated in the following section; reflecting on our research question: *Can ELECTRA achieve state-of-the-art performance in biomedical question answering when pretrained on biomedical corpora?*

A Effect of Biomedical Pretraining

Reflecting on our research question, we aim to assess whether pretraining ELECTRA on biomedical text leads to improved performance on biomedical question answering. In order to test this hypothesis, we compare results from the general-domain ELECTRA model, initialised with the

pretrained weights from Clark et al. (2020) [4], to our results for BioELECTRA. We conduct finetuning on ELECTRA to develop a comparable set of biomedical question answering models. N.B. we finetune on both general and biomedical-domain question answering to make for a fair comparison to our best-performing BioELECTRA models. We compare the highest scores of the ELECTRA and BioELECTRA small and base models (*table 9*), averaged across all batches.

Table 9: Performance comparison for ELECTRA and BioELECTRA

	Small Models		**Base Models**	
Question Type	ELECTRA	BioELECTRA	ELECTRA	BioELECTRA
Yes/No *(Macro F_1)*	0.588	0.677	0.643	0.722
Factoid *(MRR)*	0.203	0.334	0.211	0.364
List *(Avg F_1 Score)*	0.196	0.340	0.200	0.346

Table 9 shows that BioELECTRA outperforms ELECTRA on all question types when finetuned on biomedical question answering. When comparing our best-performing models from both categories - we obtain performance improvements of 12.3%, 72.5% and 73.0% for yes/no, factoid and list questions, respectively. This highlights the importance of domain-specific pretraining to achieve strong performance in downstream question answering tasks - particularly for extractive question answering. Notably, our BioELECTRA-Base models achieve comparable performance to BioELECTRA-Small on list questions, and obtain improvements of 8.98% and 12.29% on factoid and yes/no questions. However, the training time required to pretrain BioELECTRA-Base is 4.3 times greater than that of BioELECTRA-Small. This is due to the model size and the need to reduce the batch size for our larger model to fit within our memory constraints. We conclude that for list questions, the marginal improvement in performance does not justify the computational cost of pretraining ELECTRA with more parameters; however, this is worthwhile for factoid and yes/no questions.

B Solution Strengths

In this project, we demonstrate that large transformer-based biomedical question answering models can be efficiently trained to achieve competitive results, subverting the trend of using larger, more complex models with inaccessible resource requirements. We evaluate our solutions on the gold-standard testing set from the BioASQ 8B biomedical question answering challenge where we find that our BioELECTRA-Base model, finetuned on general-domain and biomedical-domain question answering, outperforms our other models. On yes/no questions, our best-performing binary classification model achieves an average Macro F_1 Score of 0.722 on yes/no questions. For factoid and list questions, our best-performing extractive model achieves an average MRR score of 0.364 and an average F_1 score of 0.346. In all three question types, we exceed the median score of biomedical QA models submitted to the 2020 BioASQ challenge on 4, 2 and 2 test batches out of 5 for yes/no, factoid and list questions, respectively.

In summary, we find that pretraining ELECTRA on biomedical text enhances downstream biomedical question answering peformance for yes/no, factoid and list questions. This is likely due to the increased similarity between the vocabulary used in the pretraining corpus and the BioASQ dataset, allowing the model to better interpret a biomedical context snippet relative to a query. We provide a performance comparison between the small and base sizes of BioELECTRA on biomedical question answering, successfully tweaking the hyperparameters for

BioELECTRA-Base to allow us to use input sequences up to a maximum length of 256 tokens within our memory constraints. In general, our BioELECTRA-Base models outperform our BioELECTRA-Small models, and we hypothesise that this is due to the ability to pass longer input sequences to our base model (256 as opposed to 128 tokens). We believe that this provides additional context to our model which enhances predictions.

C Solution Limitations

Due to the adoption of an extractive question answering technique, our solution is unable to attempt a subset of questions from the BioASQ dataset, as 26% of factoid and list questions in the testing sets are unanswerable in an extractive setting i.e. the exact answer is not provided in the context paragraph. Extractive question answering models can also suffer from position bias [9], where the model learns to predict answer spans in the first sentence of each context paragraph. Another limitation of our solution is that our evaluation method differs slightly from the evaluation method used in the BioASQ challenge. Despite using identical evaluation metrics, experts in the BioASQ challenge manually scan predictions for synonyms and abbreviations which are not automatically detected by our method. For instance, if the exact answer is *"measles"*, the term *"measles virus"* is a valid synonym, but will not be considered correct by our exact match evaluation method. This is likely to affect our results negatively, and in reference to our research question, this makes for a less accurate comparison to the state-of-the-art. By submitting our model to the 2021 BioASQ challenge, we enable experts to evaluate our model against the state-of-the-art, although the results are yet to be published for 2021.

D Ethical Considerations

In recent years, the growing reliance on large-scale datasets to maximise the performance of language models has become a serious ethical issue within NLP. As the size of training datasets increases, so does the difficulty of scrutinising their content, and poorly-curated datasets run the risk of encoding biases towards marginalised communities [1]. As part of our experimentation, we instantiate ELECTRA with the pretrained weights from the original ELECTRA paper [4]; however, the data used to obtain these pretrained weights is not available to scrutinise. They include extracts from English Wikipedia in their pretraining corpus, and this source could include lower-quality and potentially prejudiced information. While there is a higher ethical standard for articles published in medical journals, we could unintentionally encode bias into our pretraining corpus by omission. For instance, if we include an abstract detailing the efficacy of a specific treatment for men and omit comparable information for women, this could result in our model providing gender-biased information to users and must be taken into consideration before use. Furthermore, while our solution is competitive among the solutions submitted to the 2020 BioASQ challenge, it is perhaps necessary for a biomedical question answering system to provide correct answers 100% of the time if it is to be used in the field. Disseminating inaccurate health information could potentially cause serious harm to consumers, even with systems that demonstrate very high, but imperfect, performance.

VI CONCLUSIONS

The development of a robust biomedical question answering model has the potential to widen access to reliable health information. In this project, we develop a novel biomedical language

model (BioELECTRA) and evaluate its downstream performance on biomedical question answering. Furthermore, we make the pretrained weights of BioELECTRA publicly-available[4] on Huggingface [30]. Due to the difference in word distribution in biomedical and general-domain text, numerous studies have shown that domain-specific pretraining is necessary to achieve strong performance in biomedical question answering [10, 13, 29]. However, state-of-the-art language models are computationally-expensive to pretrain - hence, we harness the efficient pretraining approach of the ELECTRA model [4] to perform unsupervised pretraining on a large corpus of medical text. By finetuning BioELECTRA on the SQuAD [24], BoolQ [3] and BioASQ [16] datasets, we develop a set of binary classification and extractive question answering models to tackle yes/no, factoid and list questions.

In answer to our research question, we show that pretraining ELECTRA on biomedical text enhances downstream performance in biomedical question answering by by 12.3%, 72.5% and 73.0% for yes/no, factoid and list questions. Furthermore, we find that our BioELECTRA-Base models outperform our BioELECTRA-Small models overall, although finetuning on general-domain question answering as well as biomedical-domain question answering has a greater impact on performance than upgrading model size. Overall, our BioELECTRA-Base model, finetuned on general-domain and biomedical-domain question answering obtains the highest scores on all three question types. In comparison to the state-of-the-art, our solution does not exceed the benchmark score of the best-performing models submitted to the 2020 BioASQ challenge. However, we are confident that this performance gap will narrow when evaluated by the expert panel in the 2021 BioASQ challenge, due to their more flexible evaluation approach. We submit BioELECTRA to the question answering phase of the BioASQ challenge (2021) - the results[3] of our submission are pending and will be available in the near future. This demonstrates the potential for compute-efficient transformer-based language models to achieve competitive performance.

Given that our best-performing BioELECTRA models are finetuned on general and biomedical-domain question-answering datasets, we hypothesis that further improvements could be made by finetuning on a larger quantity of general-domain question-answering data. Another suitable direction for future work may be to apply our biomedical pretraining and finetuning procedures to a generative model, such as T5 [23], as 26% of examples from the BioASQ 8B testing sets are unanswerable in an extractive setting. Finally, we focus on the English language, however we could expand our question answering models to tackle multilingual data.

References

[1] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021.

[2] Konstantinos I Bougioukas, Emmanouil C Bouras, Konstantinos I Avgerinos, Theodore Dardavessis, and Anna-Bettina Haidich. How to keep up to date with medical information using web-based resources: a systematised review and narrative synthesis. *Health Information & Libraries Journal*, 2020.

[3] Results published at participants-area.bioasq.org/results/9b/phaseB

[3] Pretrained weights available at huggingface.co/molly-hayward

[3] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*, 2019.

[4] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020.

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

[6] Jen-Chieh Han and Richard Tzong-Han Tsai. Ncu-iisr: Using a pre-trained language model and logistic regression model for bioasq task 8b phase b. 2020.

[7] Serge PJM Horbach. Pandemic publishing: Medical journals strongly speed up their publication process for covid-19. *Quantitative Science Studies*, 1(3):1056–1067, 2020.

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[9] Miyoung Ko, Jinhyuk Lee, Hyunjae Kim, Gangwoo Kim, and Jaewoo Kang. Look at the first sentence: Position bias in question answering. *arXiv preprint arXiv:2004.14602*, 2020.

[10] Vaishnavi Kommaraju, Karthick Gunasekaran, Kun Li, Trapit Bansal, Andrew McCallum, Ivana Williams, and Ana-Maria Istrate. Unsupervised pre-training for biomedical question answering. *arXiv preprint arXiv:2009.12952*, 2020.

[11] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

[12] Md Tahmid Rahman Laskar, Xiangji Huang, and Enamul Hoque. Contextualized embeddings based transformer encoder for sentence similarity modeling in answer selection task. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5505–5514, 2020.

[13] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4), 2020.

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

[15] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[16] Anastasios Nentidis, Anastasia Krithara, Konstantinos Bougiatiotis, and Georgios Paliouras. Overview of bioasq 8a and 8b: Results of the eighth edition of the bioasq tasks a and b. 2020.

[17] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, and Chanan. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. 2019.

[18] Ioannis Pavlopoulos, Aris Kosmopoulos, and Ion Androutsopoulos. Continuous space word vectors obtained by applying word2vec to abstracts of biomedical articles. *Word Journal Of The International Linguistic Association*, pages 1–4, 2014.

[19] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014.

[20] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[21] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

[22] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[23] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

[24] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

[25] Mike Schuster and Kaisuke Nakajima. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE, 2012.

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[27] Gary M Weiss, Kate McCarthy, and Bibi Zabar. Cost-sensitive learning vs. sampling: Which is best for handling unbalanced classes with unequal error costs?

[28] Dirk Weissenborn, Georg Wiese, and Laura Seiffe. Making neural qa as simple as possible but not simpler. *arXiv preprint arXiv:1703.04816*, 2017.

[29] Georg Wiese, Dirk Weissenborn, and Mariana Neves. Neural question answering at bioasq 5b. *arXiv preprint arXiv:1706.08568*, 2017.

[30] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, 2020.

[31] Wonjin Yoon, Jinhyuk Lee, Donghyeon Kim, Minbyul Jeong, and Jaewoo Kang. Pre-trained language model for biomedical question answering. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2019.