EHReader: A Medical Healthcare Question Answering System

Jerry Li
EECS
MIT

Lawrence Wong
EECS
MIT

Yang Xiang
SEAS
Harvard

Estelle Yao
HSPH
Harvard

Abstract

Question answering (QA) is a prominent challenge in natural language processing research that requires machines to predict the correct answer to a posed question by extracting it from a given context. In some cases, QA tasks also involve determining "answerability": whether the answer is present at all in the passage. Recent research has begun to explore domain-specific QA systems, such as for usage in medical contexts. The growing adoption of electronic health records (EHR) in the healthcare system poses a specific QA challenge: retrieving answers from clinical notes to inform medical decisions.

This paper introduces the EHReader model based on the Retrospective Reader architecture. The EHReader model incorporates quick reading and deep reading modules, enabling it to evaluate answerability and then verify the answer more comprehensively quickly. We compare EHReader to baseline DistilBERT and BioBERT models for medical QA tasks. The proposed model incorporating only the QuickReader module achieves state-of-the-art results on the benchmark EmrQA medical dataset and outperforms the baseline DistilBERT and BioBERT models.

1 Introduction

Question answering (QA) in natural language processing consists of tasks regarding programming the system to represent questions to provide adequate answers by information retrieval, such as identifying relevant passages from documents, websites, or even generating answers (Mutabazi et al., 2021). The integrative task of QA requires models to conduct sentiment analysis, correct information extraction, text summarization, and natural language generation (Piskorski and Yangarber, 2013; Sankarasubramaniam et al., 2014). With the development of domain-specific QAs, we have seen QAs applied to educational settings where reliable answers are extracted from documents to assist student questions (DERİCİ et al., 2018).

In recent years, medical-specific QA systems have also gained research attention. With the rapidly growing adoption of electronic health records (EHR) in the American healthcare system, clinicians frequently rely on retrieving answers from the system to inform clinical decisions. EHR is defined as longitudinal medical records from "unstructured clinical notes" and "structured vocabularies" (Pampari et al., 2018). However, the non-uniformity of record types across systems poses enormous challenges for health care workers to obtain relevant and accurate answers in real-time (Tayefi et al., 2021).

While the state-of-the-art general QA model based on Bidirectional Encoder Representations from Transformers (BERT) and DistilBERT, the lightweight counterparts, has shown promising results and advancement in the field, there has been emerging research focusing on disentangling the complexity embedded in EHR (Sanh et al., 2020). This research includes identifying domain-specific terminologies and modeling implicit relationships between question-answer pairs that may not be easily recognizable (Devlin et al., 2019). In addition, recent work such as BioBERT has shown improvement in biomedical relation extraction (Lee et al., 2019).

Given the pressing need for credible EHR-based QA systems and space for creative modeling design, we aim to leverage large-scale publicly available medical QA data to build upon existing medical QA models into a more robust and advanced system with higher predictive accuracy.

2 Related Work

Medical question-answering systems (MQAS) that combine linguistic insights with machine learning architectures to capture information from complex medical notes tend to perform better than tradi-

tional methods. For example, Dai, et al. proposed an inception convolutional autoencoder model solution for Chinese healthcare question-answer data (Dai et al., 2021). Autoencoders compress input data into an encoded representation to address common issues such as high dimensionality, sparseness, noise, and nonprofessional expression related specifically to medical datasets. On the other hand, Zhang, et al. proposed a hybrid model that combines CNN with GRU to solve a similar task. This approach combines CNN's ability to extract and reduce dimensionalities and RNN's ability to capture sequential relationships (Zhang et al., 2020a).

There also exist generic QA models explicitly built for the Stanford Question Answering Dataset (SQuAD) (Rajpurkar et al., 2018). Although this is not a medical dataset, we can leverage ideas from architectures that achieve high EM and F1 scores on the SQuAD dataset. For example, BERT-based models have demonstrated strong performance on the SQuAD dataset and various other NLP tasks. BERT uses the transformer encoder portion to leverage bidirectional training and generate a language model. Specifically, it is trained toward two objectives: (1) Masked LM, in which BERT is trained to predict masked words in the input sequence, as well as (2) Next Sentence Prediction, in which it predicts whether a given sentence follows another. BERT can be applied toward QA tasks by using the context and question concatenated as input and learning a start and end token vector to predict the answer's location within the context. BERT models can be tuned for application in medical contexts. For example, BioBERT, trained on biomedical corpora, outperforms BERT on domain-specific text mining tasks (Lee et al., 2019). Researchers have also fine-tuned BERT specifically for medical QA by training on the EmrQA dataset, to further success (Wen et al., 2020). In recent years, more advanced model architectures have significantly improved state-of-the-art performance on SQuAD. For example, Retrospective Reader uses a transformer-style multi-head cross attention to learn a question-aware context representation to determine the answer span within a passage (Zhang et al., 2020b).

Our work will leverage architecture ideas built for the SQuAD dataset and replicate them for healthcare datasets like EmrQA. Likewise, we will experiment with domain-informed representations that motivated the architectures for medical question-answering systems for Chinese medical datasets. We will incorporate a pre-trained embedding function specific to medical terminologies. Ultimately, we plan to adapt, fine-tune, and improve upon the Restrospective Reader model for applications on healthcare datasets.

3 Background

Electronic health records (EHR) are essential resources to help physicians make clinical decisions and tailor treatments based on the patient's conditions (Ely JW). Physicians often review EHR records to seek answers to their questions and enhance patient care quality. Moreover, the rapidly growing adoption of EHR in the American healthcare system means that physicians spend more time reviewing these documents. Physicians usually have limited time to digest the information thoroughly. For example, a study found that physicians often abandon seeking an answer to a question if the search for the answer in the EHR records takes longer than two minutes (Alper BS). Also, another study showed that it could take a healthcare provider more than 30 minutes to search for an answer from EHRs (Hersh WR).

Recent work has focused on improving healthcare tailored question-answering language models to enhance healthcare quality and best utilize healthcare workers' time for treatment rather than administrative searching for information. However, with the complexity of medical data that is hugely different from existing pre-trained language models and the unstructured nature of EHR data collated from various sources, medical QA models still warrant improvements and research endeavors.

4 Method

We intend to develop and train models toward the SQuAD/EmrQA-style objective: given a context and question, predict the answerability of the question and answer span within the context. As a baseline model, we plan to fine-tune a DistilBERT QA model by training it on medical datasets such as EmrQA. The Hugging Face transformers library includes a pre-built, pre-trained (on SQuAD) BERT model for QA. Adapting this model gives us insight into how we might fine-tune the Retrospective Reader model for medical QA and serves as a useful performance benchmark.

4.1 DistilBERT

A baseline DistilBERT QA model is fine-tuned on the EmrQA dataset. DistilBERT runs 60% faster and requires 40% less parameters than the original BERT model while preserving over 95% of the performance (Sanh et al., 2020). The DistilBERT model uses a multi-layer bidirectional transformer encoder for its architecture similar to the BERT model (Devlin et al., 2019). DistilBERT's input is tokenized passage-question pairs separated by a [SEP] token. In addition, a learned embedding is added to each token to denote which sentence it belongs to. Finally, the input representation for each token is given simply by the sum of the token, segment, and position embeddings.

4.1.1 Pre-training DistilBERT

DistilBERT is pre-trained using two unsupervised tasks: Masked Language Modeling (MLM) and Next Sequence Prediction (NSP).

Masked Language Modeling. To train a bidirectional model, the masked LM task involves randomly masking 15% of the tokens in the input sequence, then predicting the masked words given the context. Finally, the predicted word is given simply by an output softmax over the vocabulary for the final hidden vectors of the masked tokens.

Next Sentence Prediction. The following sentence prediction objective allows DistilBERT to better learn relationships between multiple sentences. In training, DistilBERT is fed a packed input sequence of two distinct sentences, where 50% of the time, the second sentence directly follows the first sentence in the original document. The other 50% of the time, the second sentence is chosen randomly from the corpus; the objective is to distinguish between the two. The prediction is made using an output layer on top of the final hidden vector of the first input token.

4.1.2 Fine-tuning DistilBERT for QA

For question-answering tasks, we pack the input question and passage into a single sequence as the input to the BERT model. The question tokens use the embedding for Sentence A, and the passage tokens use the embedding for Sentence B. Then, for fine-tuning on the output side, we introduce a start vector S and an end vector E. We then compute the probabilities of each word i being the start of the answer span by taking the dot product between the final word embedding and the start vector, followed by a softmax (and similarly for the end vector).

$$P_i = \frac{e^{S \cdot T_i}}{\sum_j e^{S \cdot T_j}} \quad (1)$$

The objective function is simply the sum of the log-likelihoods for the correct start and end tokens. For prediction, we score each candidate span from (i, j) using the formula $S \cdot T_i + E \cdot T_j$, and select the highest scoring span as the answer.

We used the BertForQuestionAnswering model in the Hugging Face transformers library - which already implements, pre-trains, and fine-tunes BERT on the SQuAD dataset - as the basis for our BERT model, which we then trained on the EmrQA dataset.

4.2 BioBERT

BioBERT (Bidirectional Encoder Representations from Transformers for Biomedical Text Mining) is a pre-trained, domain-specific language representation model for medical data (Lee et al., 2019). BioBERT uses virtually the same model architecture as BERT; however, it is pre-trained on large-scale biomedical corpora. Specifically, BioBERT is initialized with the weights from BERT, is then pre-trained on PubMed abstracts and PMC articles, and is finally fine-tuned on three text mining tasks: named entity recognition (NER), relation extraction (RE), and question answering (QA). In literature, BioBERT has been shown to outperform BERT in all three tasks significantly. For this reason, in addition to DistilBERT, we also fine-tuned a BioBERT model on the EmrQA dataset to compare its performance and potentially serve as an additional benchmark.

4.3 EHReader

The model implemented from scratch is based on Retrospective Reader (Zhang et al., 2020b). This model imitates how people answer a comprehensive question by going through the passage quickly to get the main idea and then reading the passage carefully to verify the answer. Retrospective Reader does this by using the sketchy reading module to evaluate the answerability of the question and the intensive reading module to predict the answer span, which is to recognize the start and end span of a matching answer to the question. These models are trained separately from one another. Like the sketchy reading module, our QuickReader module will use DistilBERT/BioBERT to embed the question and passage into token embeddings, position

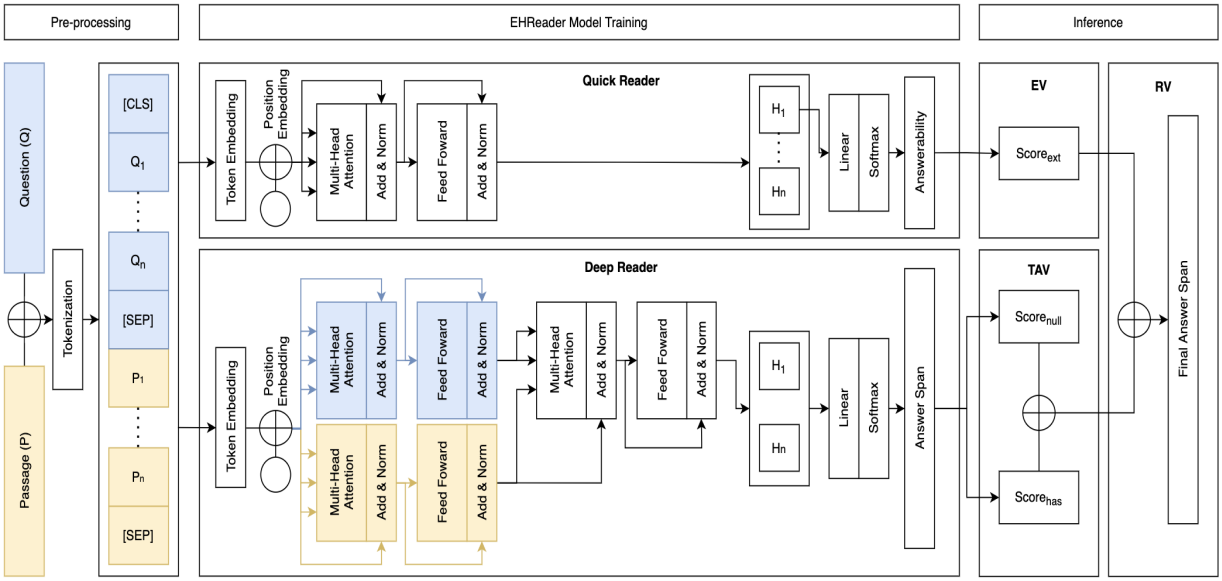


Figure 1: The entire workflow includes pre-processing, model architecture, and scoring criteria. The EHReader architecture consists of Quick Reader and Deep Reader. The Quick Reader is composed of a linear layer and softmax. In contrast, the Deep Reader is composed of two self attentions on passage and question separately and cross attention on both, followed by a feed-forward neural network.

embeddings, and token-type embeddings. Then, it will use a multi-layer transformer to learn contextual representations of the interaction layer. Finally, the module produces a probability of answerability using the embedding of the first token and a softmax function paired with a negative log-likelihood loss function. In parallel to the Quick Reader module, we aim to implement a Deep Reader module similar to the intensive reading module that performs matching attention and cross attention between question and context to learn a question-aware context representation (Chen et al., 2021). This representation is used to predict probabilities of answer span within the context. An unanswerable question-passage pair is represented by the first index corresponding to the [CLS] token of the start and end probability vector. A weighted sum of the negative log-likelihood of the start and end tokens for the matching answer is used as a loss function for optimization. The architecture of EHReader is given in Figure 1.

4.3.1 Quick Reader

The quick reader aims to generate a score that measures the answerability of the given passage-question pair. After tokenizing and embedding the passage-question pair, we feed the embedding into a multi-layer attention transformer block. After that, we apply a feed-forward neural network followed by a softmax function on the first embedding of the first token to predict the classification logits $\hat{y}_i$. Finally, we use cross-entropy as the training objective given as follows:

$$L^{ans} = -(y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i)) \quad (2)$$

As inspired by Retrospective Reader, the external verification (EV) score is calculated by

$$score_{ext} = logit_{na} - logit_{ans} \quad (3)$$

such that $logit_{na}$ is the unanswerable logit and $logit_{ans}$ is the answerable logit.

4.3.2 Deep Reader

The deep reader aims to give an answerability and the span prediction. Therefore, it employs the same encoding and interaction procedure to obtain the representation H. The difference is that the embedding is split into question and passage embedding blocks by the position information. Self-attention is performed on each block, and cross attention is used to generate one final question-passage-aware embedding. The final embedding is passed as input of a forward neural network to obtain the vector of start and end probabilities, s and e. The training objective of answer span prediction is defined as cross-entropy loss for the start and end predictions:

$$L^{span} = -(log(p^s_{y^s_i}) + log(p^e_{y^e_i})) \quad (4)$$

where y^s_i and y^e_i are respectively ground-truth start and end positions of example i.

As inspired by the Retrospective Reader, we apply Threshold-based Answerable Verification (TAV) to compute the verification score for the deep reader. Given the output start and end probabilities s and e, we calculate the has-answer score $score_{has}$ and the no-answer score $score_{null}$, given by

$$\begin{aligned} score_{has} &= max(s_k + e_l), \quad 1 < k \leq l \leq n \\ score_{null} &= s_1 + e_1 \end{aligned} \tag{5}$$

The final no-answer score is defined by difference between $score_{has}$ and $score_{null}$, i.e. $score_{diff} = score_{has} - score_{null}$. An answerable threshold δ is determined according to the train and validation split. Our model predicts the answer span if $score_{diff} < \delta$ and null string otherwise.

4.3.3 Rear Verification (RV)

As inspired by the Retrospective Reader, the rear verification (RV) combines EV and TAV for the final answer. This inference methodology aims to reduce the number of false-positive predictions.

$$v = \beta_1 score_{ext} + \beta_2 score_{diff} \tag{6}$$

where β_1 and β_2 are weights. Similar to TAV, our model predicts the answer span if $v < \delta$ and null string otherwise.

5 Experimental Setup

5.1 Setup

We used available pre-trained BERT architecture to build a baseline QA model for each dataset (medication and relations) separately. The question and passage pair is tokenized using a pre-trained tokenizer with a maximum length of 512. Then, we fine-tuned the DistilBERT (*distil-base-uncased*) and bioBERT (*dmis-lab/biobert-v1.1*) pre-trained model using Adam optimizer with decoupled weight decay regularization, a learning rate of 3e-5, and a batch size of 8. Each dataset was divided into a 60-20-20 train-validation-test split. The termination epoch was dynamically determined when the loss on the validation set began increasing relative to the average of the last three validation losses.

Quick reader and deep reader from the EHReader architecture were trained in parallel, similar to the baseline model. Quick reader optimizes for answerability while deep reader (and baseline BERT) models optimize for answer span predictions. The TAV threshold for deep reader-like models and RV threshold for the quick reader and any deep reader-like model pair were finetuned based on the evaluation metric using the train and validation split. Finally, all models were trained on the GTX 1070 GPU graphics card with 8GB of VRAM.

5.2 Benchmark Datasets

EmrQA The primary datasets we used to train and evaluate our models are EmrQA that hosts over 400,000 question-answer pairs, all semi-automatically generated from the medical domain. Specifically, questions in the EmrQA dataset are generated from annotated question templates, which are then populated (along with their corresponding answers) using existing annotations on clinical notes (Pampari et al., 2018). The EmrQA dataset contains six topics: medical relations, medications, heart disease, obesity, and smoking.

To train a question-answering model that predicts the start and end span of the question in the context, we have omitted the heart disease and obesity datasets, which have only yes/no answers. We have also omitted using the heart disease dataset in training because of its lack of structure in preprocessing steps noted by the authors. Thus, our training focuses on using medical relations and medications datasets (Table 1). Each dataset contains three pieces of information: question, the exact answer, and clinical notes, which is the context from which the exact answer is extracted. The key differences between these two datasets lie in their question properties and the type of medical terms. The medication data set ask questions regarding patients' medication history with forms akin to "has the patient taken [medication]" or "what is the does of [medication]". The relation data set focuses on patients' past diagnosis and symptoms with question forms of "does the patient have [symptom]" or "what is the diagnosis for [symptom]." Thus, using both datasets can help us understand how models are learning different medical terms and question types.

With the objectives to train a question-answering model that predicts correct answer spans and a question's answerability, we have created negative samples from the two datasets mentioned above. To do so, we tokenized each clinical note by NLTK pre-trained tokenizer and limited the length of each sub-clinical note to be fewer or equal to length 512,

Dataset	# QA pairs	# Clinical notes	Post-processing data	Question type
Medication	255,908	261	Train : 36,288 Validation : 12,096 Test: 12,096	Has the patient taken [medication]? What is the does of [medication]?
Relations	141,243	424	Train : 79,900 Validation : 26,634 Test: 26,634	Does the patient have [symptom] ? What is the diagnosis for [symptom]?

Table 1: Summary of Medication and Relations data sets used for all three model training and evaluations. QA pairs and clinical notes are raw values from the database, and post-processing data contains QA pairs used in the experiments.

which is the length of input for BERT pre-trained models and the Retrospective Reader model. Thus, if the sub-clinical note contains actual answers, the question given the context is answerable; conversely, if the sub-clinical note does not contain the answer, the question given the context is not answerable. We further down-sampled unanswerable question-answer pairs to balance the number of answerable and unanswerable pairs in both datasets (Table 1).

5.3 Evaluation

As with many QA models and datasets in literature, we will use the exact match (EM) and F1 metrics to evaluate our model. The EM score is an 'all-or-nothing' metric that calculates the percentage of the model's predictions that perfectly match the ground truth answer. In contrast, the F1 score is a softer metric that measures the average overlap between the prediction and ground truth answer at the token level. The EM and F1 benchmarks of many state-of-the-art models are readily available on the SQuAD leaderboard for comparison. Perhaps more importantly, we compare our scores against those achieved by other models on domain-specific (i.e., medical) datasets such as EmrQA. Finally, we also fine-tune our baseline models and evaluate our EHReader model against baseline models' performance.

6 Results & Analysis

The fine-tuned distilBERT model without threshold-based answerable verification outperformed the distilBERT with frozen weights (except for the classifier) based on the EM and F1 scores on the test split. Furthermore, the frozen weight model converged on a local optimum that predicts the null answer span for all samples. We also compared the baseline distilBERT with baseline bioBERT without answerability and found that bioBERT did not outperform the baseline model. This decrease in performance suggests that using a biological embedding might not be most suitable for learning embeddings of medical question-passage pairs.

To test the hypothesis of whether including TAV can improve the model's ability to identify correct question-answer span, we compared the baseline distilBERT and bioBERT model with and without TAV. For both models, the inclusion of TAV gained non-trivial EM and F1 scores, thus showing the effectiveness of imposing an extra constraint on the model to improve its prediction precision. Furthermore, RV's answerability metric further boosts the EM and F1 scores, as seen in the QuickReader-distilBERT(+RV) experiment.

To test whether a cross attention layer can enhance question-answer prediction, we benchmarked our EHReader (Deep Reader paired with Simple Reader module) with the QuickReader-distilBERT(+RV) model. The results showed that our EHReader model encountered vanishing gradients during training and made null predictions for every sample.

7 Discussion

According to the results, the QuickReader-distilBERT(+RV) appears to be the best model and outperforms the baseline models. On the other hand, the EHReader(+RV) does not perform well, and it predicts all negative predictions since we encounter a gradient vanishing problem in training the model. We plan to use some techniques to deal with the gradient vanishing problem in the future. In addition, we can see that inclusion of TAV is beneficial as both baseline distilBERT and bioBERT perform better. We also hypothesized that bioBERT model did not perform as well as the baseline distilBERT because the model was originally trained on binary questions rather than an-

Model	Medication				Relations			
	Dev		Test		Dev		Test	
	EM	F1	EM	F1	EM	F1	EM	F1
Our Implementation inspired by Retro-Reader (Zhang et al., 2020b)								
EHReader(+RV)	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
QuickReader-distilBERT(+RV)	94.3	95.8	**86.5**	**89.6**	97.1	97.4	**94.6**	**95.2**
distilBERT (Sanh et al., 2020)								
distilBERT (Frozen)	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
distilBERT	93.5	95.0	85.2	88.1	96.7	97.0	94.0	94.6
distilBERT(+TAV)	94.3	95.7	**86.4**	**89.3**	97.1	97.4	**94.4**	**95.0**
bioBERT (Lee et al., 2019)								
bioBERT	63.2	64.5	59.3	60.8	48.1	50.9	46.4	49.2
bioBERT(+TAV)	65.8	67.5	**60.9**	**63.0**	59.6	60.0	**51.2**	**53.1**

Table 2: The results for EmrQA medication and medical relations datasets (all numbers are percentages).

swering span-oriented questions in our case. Also, the original paper pointed out the model's ability to recognize medical terms yet may not be as good at question-answering tasks (Lee et al., 2019).

7.1 Comparison of Predictions

To gain a better intuitive understanding of the differences in our model predictions, we give several prediction examples from the best model of each type: QuickReader-distilBERT(+RV), distilBERT(+TAV), and bioBERT(+TAV).

The first apparent empirical observation is that bioBERT often predicts false positives. In contrast, the other two models predict that those examples are answerable and give the correct gold truth answer. Therefore, one plausible contribution toward bioBERT's low EM/F1 scores is its tendency to produce false positives, even with threshold-based answerable verification.

Figure 2A shows a different example where bioBERT gives an answer span prediction, but it does not match the gold truth, while both the distilBERT and QuickReader-distilBERT models are correct. For this question, bioBERT appears to better recognize the medical term "nitroglycerin 1/150" and is the only model whose answer span includes the phrase. Still, it fails to recognize the importance of the phrase "chest pain." This example reflects results seen in literature, where bioBERT can better identify biomedical keywords but still struggles with general question-answering due to its specific fine-tuning tasks.

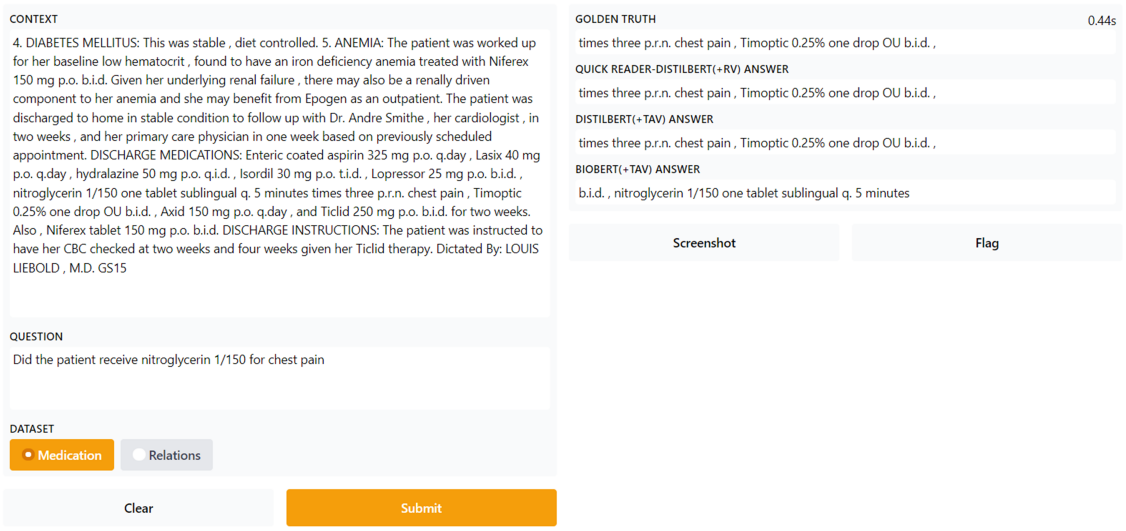
Another important observation is that the QuickReader-distilBERT and distilBERT models frequently give the same predictions. For example, in Figure 2B, we present an example in which they both predict the same incorrect answer span. The question is challenging to correctly answer since it relies on the model understanding the difference between taking medication and potentially taking one. Hence, the error made by both models is easily explicable.

Finally, we present a rare example in which the distilBERT and QuickReader-distilBERT models have different predictions. In Figure 2C, distilBERT incorrectly predicts unanswerable, while QuickReader-distilBERT produces the correct answer span. We also identified other examples of the same predictions, but the question is unanswerable. Jointly, these examples show a tendency for distilBERT to predict unanswerable more frequently than QuickReader-distilBERT, leading to both more false positives and true negatives. Hence, the improvement in EM/F1 scores for QuickReader-distilBERT may be ascribed to a more significant reduction in false positives.

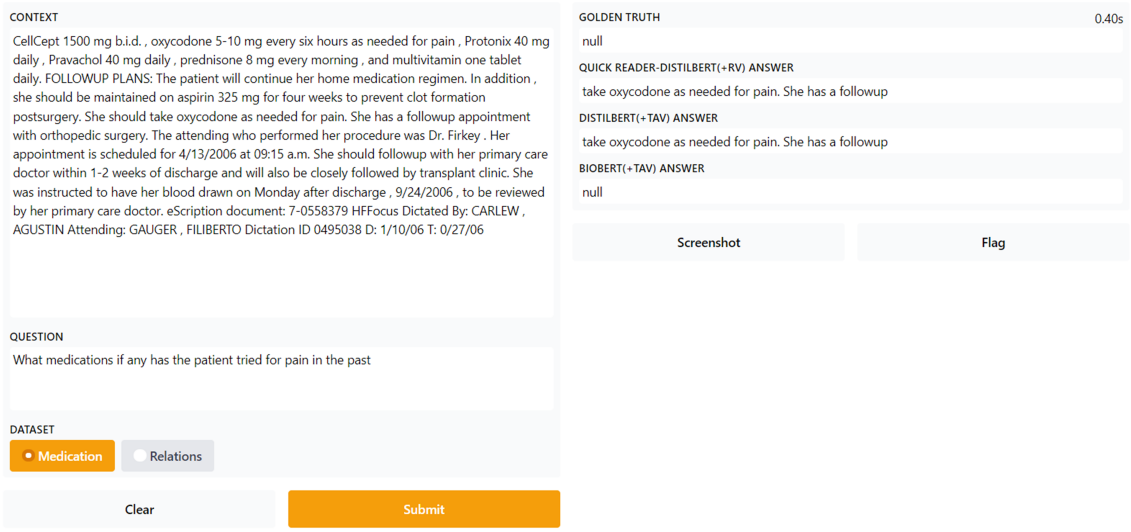
7.2 Computational Limitations

Initially, we ran into out-of-memory issues in the training of BERT models. This problem occurred in Colab Pro using a P100 GPU, and decreasing the batch size did not help. Training on T4 TPU was possible, but the runtime was on the orders of days per epoch. In the end, we got access to a GTX 1070 graphics card with 8GB of VRAM and were able to train on GPU with a batch size of 8. This batch size could be increased if we freeze the weights within the BERT model in the fine-tuning task.

A



B



C

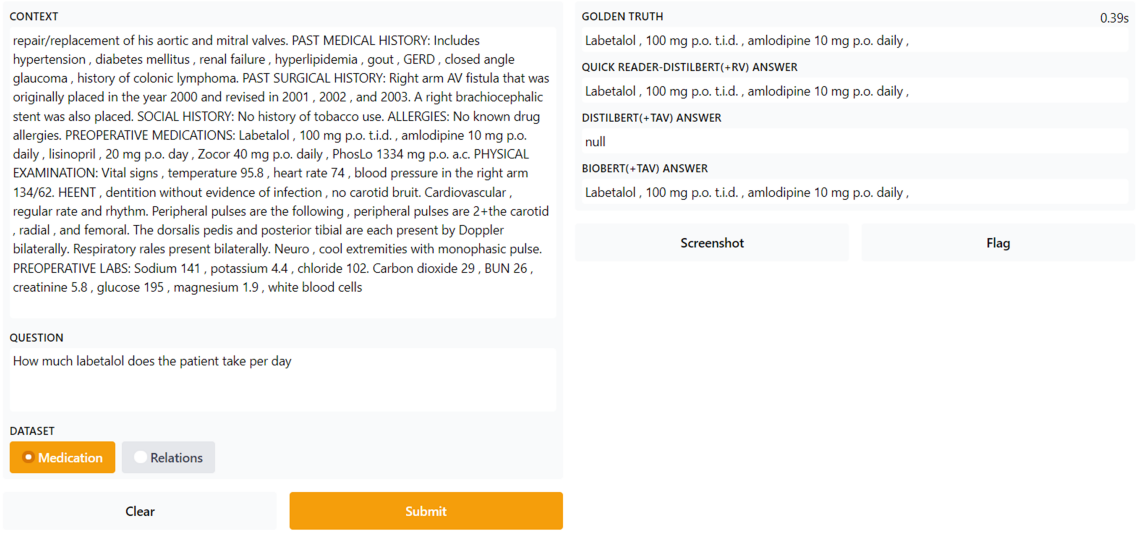


Figure 2: Answer prediction examples from the distilBERT(+TAV) baseline, bioBERT(+TAV), and QuickReader-distilBERT(+RV) models.

8 Conclusion

As the usage of electronic health records continues to grow in the healthcare system, so does the interest in developing a machine capable of predicting answers to medical questions by obtaining them from clinical notes. This paper presented the EHReader model inspired by the Retrospective Reader architecture. EHReader can effectively predict answerability and answer span for clinical QA tasks with both quick reading and deep reading components when evaluated on the benchmark EmrQA medical dataset. Furthermore, with only the Quick Reader module, the proposed model achieved state-of-the-art results and outperformed the baseline DistilBERT and BioBERT models, demonstrating the effectiveness of the EHReader architecture for medical QA.

In the future, we will explore more techniques that deal with the gradient vanishing problem during model training. We did encounter a gradient vanishing problem when training the EHReader. The Deep Reader architecture is too complex and results in small gradient propagation. Therefore, only the parameters of Quick Reader get updated during the training process. Also, we will investigate the performance of the EHReader model on other benchmark healthcare datasets aside from EmrQA, such as MedQA and HONQA, which may involve accepting more non-standard form questions. Furthermore, we also intend to explore other state-of-the-art model architectures on the SQuAD dataset, adapt them to medical QA tasks, and evaluate their performance against the EHReader architecture.

Acknowledgments

L. W. X. Y. thank the staff members of the MIT course *6.864 Natural Language Processing* for their guidance. Special thanks go to Pranav Krishna and Joe O'Connor for their suggestions and feedback on this project.

References

White DS Ewigman Alper BS, Stevermer JJ. Answering family physicians' clinical questions using electronic medical databases.

Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. 2021. Crossvit: Cross-attention multi-scale vision transformer for image classification.

Dan Dai, Juan Tang, Zhiwen Yu, Hau-San Wong, Jane You, Wenming Cao, Yang Hu, and C. L. Philip Chen. 2021. An inception convolutional autoencoder model for chinese healthcare question clustering. *IEEE Transactions on Cybernetics*, 51(4):2019–2031.

CANER DERİCİ, YİĞİT AYDIN, ÇİĞDEM YENİALACA, NİHAL YAĞMUR AYDIN, GÜNİZİ KARTAL, ARZUCAN ÖZGÜR, and TUNGA GÜNGÖR. 2018. A closed-domain question answering framework using reliable resources to assist students. *Natural Language Engineering*, 24(5):725–762.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Ebell MH Bergus GR Levy BT Chambliss ML Evans ER Ely JW, Osheroff JA. Analysis of questions asked by family doctors regarding patient care.

Hickam DH Sacherek L Friedman CP Tidmarsh P Mosbaek C Kraemer D Hersh WR, Crabtree MK. Factors associated with success in searching medline and applying evidence to answer clinical questions.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

Emmanuel Mutabazi, Jianjun Ni, Guangyi Tang, and Weidong Cao. 2021. A review on medical textual question answering systems based on deep learning approaches. *Applied Sciences*, 11(12).

Anusri Pampari, Preethi Raghavan, Jennifer Liang, and Jian Peng. 2018. emrqa: A large corpus for question answering on electronic medical records.

Jakub Piskorski and Roman Yangarber. 2013. *Information Extraction: Past, Present and Future*, pages 23–49. Springer Berlin Heidelberg, Berlin, Heidelberg.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Yogesh Sankarasubramaniam, Krishnan Ramanathan, and Subhankar Ghosh. 2014. Text summarization using wikipedia. *Information Processing & Management*, 50(3):443–461.

Maryam Tayefi, Phuong Ngo, Taridzo Chomutare, Hercules Dalianis, Elisa Salvi, Andrius Budrionis, and Fred Godtliebsen. 2021. Challenges and opportunities beyond structured data in analysis of electronic health records. *WIREs Computational Statistics*, 13(6):e1549.

Andrew Wen, Mohamed Y Elwazir, Sungrim Moon, and Jungwei Fan. 2020. Adapting and evaluating a deep learning language model for clinical why-question answering. *JAMIA Open*, 3(1):16–20.

Yuteng Zhang, Wenpeng Lu, Weihua Ou, Guoqiang Zhang, Xu Zhang, Jinyong Cheng, and Weiyu Zhang. 2020a. Chinese medical question answer selection via hybrid models based on cnn and gru. *Multimedia Tools and Applications*, 79(21):14751–14776.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2020b. Retrospective reader for machine reading comprehension.