A Reproduction Study of SleepQA dataset for Extractive Question Answering

Qi Zhou
{qizhou8}@illinois.edu

Group ID: 89
Paper ID: 57
Presentation link: https://youtu.be/86V2p8Kuv3A
Code link: https://github.com/Qi9726/DLH_reproduction_study

1 Introduction

Chronic illnesses account for a significant proportion of global deaths and place a burden on healthcare systems (Bojic et al., 2022). Primary prevention, which aims to manage the causes and risk factors of diseases to prevent them from occurring, has gained increasing attention, and health coaching is a recognized and effective strategy for preventing chronic diseases through primary prevention (Yang et al., 2020).

The main objective of the SleepQA paper (Bojic et al., 2022) is to create a Question Answering (QA) system that is tailored to the healthcare industry, with the aim of assisting health coaches in their efforts towards preventative care. Specifically, the authors focus on sleep coaching and introduce their domain-specific dataset named sleepQA to create a health coaching model that provides precise answers to factual sleep-related queries, thus aiding health coaches in their work.

The work of this study is to replicate the authors' approaches of fine-tuning domain-specific BERT models using the SleepQA dataset and assess the dataset's usefulness as presented in the original paper for both passage retrieval and reading tasks, which are two fundamental components of a QA system.

2 Scope of reproducibility

In an extractive (QA) pipeline, there are typically two stages involved: Passage retrieval and Passage reading. The first stage involves retrieving pertinent passages from a large textual corpus that are likely to contain the answer to a given question. The second stage involves utilizing these passages to extract the precise answer span to the question (Zhu et al., 2021).

QA systems can be classified into two categories, namely open-domain systems and domain-specific systems. Open-domain systems are typically trained on vast general domain corpora but may struggle to generalize well beyond their training domains. On the other hand, domain-specific systems often encounter difficulty due to the lack of large domain-specific datasets, which are costly and time-consuming to create and require the expertise of domain experts (Bojic et al., 2022).

To this end, the authors utilize their domain-specific SleepQA dataset to fine-tune on BERT models and adopt an approach of fine-tuning the retrieval models and reader models independently and selecting the best fine-tuned retriever and reader to create the best performing QA pipeline.

The authors' experiments show that when testing on SleepQA test set, while for retrieval task, none of the five fine-tuned domain specific BERT models achieve better results than Lucene BM25 on recall@k metric, which potentially could be attributed to the high similarity between the questions and passages, their fine-tuned reader counterparts do surpass BERT SQuAD2 model when evaluating on EM and F1 scores.

This study will run experiments to access the authors' results of fine-tuning domain-specific BERT models on the SleepQA dataset for both passage retrieval task and reading task, as well as to evaluate the effectiveness of the SleepQA dataset.

Additionally, this study created an augmented dataset named SleepQA3x based on the original SleepQA dataset. To achieve this, this work will use paraphrasing method to expand the question and passage pairs in SleepQA train set to three times its original size. The original train set consists of 4,000 question and passage pairs, whereas the augmented dataset includes 12,000 pairs. The goal is to evaluate the effectiveness of the augmented dataset in enhancing retrievers' performance.

2.1 The claims from the original paper:

- Retriever

  The authors' best fine-tuned retrieval model using PubMedBERT can only reach a score of 0.42, which is lower than the score of 0.61 achieved by Lucene BM25. The authors propose that this under-performance of their retrieval models may be attributed to the high degree of similarity between questions and passages, as the annotators could have potentially formulated questions with similar phrases from the corresponding passages. This observation is supported by their study of SleepQA dataset against other similar datasets.

- Reader

  All of the authors' five domain-specific reading models outperform the BERT SQuAD2 used as a baseline, which has an EM score of 0.5 and an F1 score of 0.64. The BioBERT BioASQ model is the best performer, achieving an EM score of 0.61 and an F1 score of 0.73. The results of the authors' reading tasks demonstrate the benefits of using SleepQA to fine-tune the BERT models.

3 Methodology

3.1 Model descriptions

The original paper uses BERT base model [1] and five domain-specific BERTs, namely BioBERT[2], BioBERT BioASQ[3], ClinicalBERT[4], SciBERT[5] and PubMedBERT[6] and build upon the framework of Dense Passage Retrieval (DPR)(Karpukhin et al., 2020).

DPR employs the concept of contrastive learning as training objective to learn sentence embedding space where sentences with similar semantics are situated close to each other, while dissimilar ones are positioned farther apart (Hadsell et al., 2006). In the framework, questions and their corresponding relevant passages are classified as positive pairs, while irrelevant passages are treated as negative pairs. It implements contrastive learning to minimize the embedding distance between positive pairs and maximize it for negative pairs, using a similarity score derived from the dot product. The negative log likelihood of the positive passage is optimized against a set of negative passages such that positive pairs have a higher degree of similarity than the negative ones (Karpukhin et al., 2020). A BERT-based dual-encoder is used to encode the question and passage separately, and then take the embeddings for the [CLS] token to perform dot products for similarity score.

3.2 Data descriptions

The SleepQA dataset was created based on two web pages[7] to collect high-quality, evidence-based, and medically reviewed sleep health related articles. From these articles, the authors extracted 7005 passages, each containing 100 to 150 words. Out of these, 5000 passages were chosen at random for the purpose of manually creating questions and their corresponding answers. To do so, five medical students were recruited as annotators and given instructions to read and understand the passages, and then generate questions and identify text spans as answers within the passages. All questions created by the annotators begin with one of the six words: who, what, where, when, why, or how, which are followed by a question mark.

The orginial dataset has been made available to the public in the authors' GitHub repository [8], where a data folder has been created containing train (4,000 samples), dev (500 samples), and test (500 samples) dataset, as well as the whole sleep corpus.

The SleepQA3x dataset in this work is created using Google's PEGASUS model [9] to paraphrase each question-passage pairs within SleepQA train set of 4,000 samples twice, resulting an augmented train dataset of 12,000 pairs. Table 1 presents a comparison of the average number of words in passages and questions for both datasets.

[1] https://huggingface.co/bert-base-uncased

[2] https://huggingface.co/dmis-lab/biobert-v1.1

[3] https://huggingface.co/gdario/biobert_bioasq

[4] https://huggingface.co/emilyalsentzer/Bio_ClinicalBERT

[5] https://huggingface.co/allenai/scibert_scivocab_uncased

[6] https://huggingface.co/microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext

[7] https://www.sleepfoundation.org and https://thesleepdoctor.com

[8] https://github.com/IvaBojic/SleepQA

[9] https://huggingface.co/tuner007/pegasus_paraphrase

Dataset	Passage	Question
SleepQA	120.5	9.9
SleepQA3x	97.7	9.1

Table 1: Comparison of average number of words in passage and question in train sets of original SleepQA and augumented SleepQA3x.

3.3 Hyperparameters

Throughout the experiment, all hyperparameters for retriever (see Table 2) and reader (see Table 3) were maintained consistent with those used in the original paper, with exceptions: 1. The batch size and dev batch size for both retriever and reader. The adjustment was made due to limited memory capacity, as elaborated in the Computational Requirements section below; 2. learning rate for retriever was adjusted to a smaller rate than the original paper (1e-5) in accordance of the smaller batch. It was noted that a high learning rate with a small batch size could readily result in overfitting.

Hyperparameter	Value
batch size †	3
dev batch size †	3
learning rate †	2e-6
eval per epoch	1
adam eps	1e-8
adam betas	(0.9, 0.999)
max grad norm	1.0
log batch step	100
train rolling loss step	100
weight decay	0.0
warmup steps	2e-6
gradient accumulation steps	1
num train epochs	30
hard negatives	0
other negatives	1

Table 2: Hyperparameters of retrieval model fine-tuning for BERTs. Symbol †indicates that the hyperparameters used in this study are different from those of the original paper.

3.4 Implementation

This replication study follows the same methodology as the original authors by utilizing the DPR codebase, which includes a retriever and a reader, and employs the codebase by Facebook available on GitHub[10]. DPR codebase is incorporated in a sophisticated framework with hydra-based configuration. To fine-tune the retrieval and reader model, this study needs only to modify the configuration in the .yaml files to run different experiments accordingly.

Hyperparameter	Value
batch size †	3
dev batch size †	3
learning rate	1e-5
eval step	500
adam eps	1e-8
adam betas	(0.9, 0.999)
max grad norm	1.0
log batch step	100
train rolling loss step	100
weight decay	0.0
warmup steps	2e-6
gradient accumulation steps	1
num train epochs	30

Table 3: Hyperparameters of reader model fine-tuning for BERTs. Symbol †indicates that the hyperparameters used in this study are different from those of the original paper.

Additionally, to ensure comparability, the study also uses the evaluation scripts[11] provided by the authors.

With regard to the generation of the augmented dataset, this study develops its own code to increase the size of the original SleepQA dataset while maintaining the same format, making it compatible with the DPR framework. The study then proceeds to fine-tune the PubMedBERT model with the new SleepQA 3x dataset for retrieval and compare the recall@1 score with those obtained from the original dataset.

3.5 Computational requirements

The hardware resources available for conducting this reproduction study is a single NVIDIA 3060 Ti GPU that has a memory capacity of 8 GB.

To fine-tune the BERT models, the first step is to load the model into memory. Loading a BERT model typically takes over 6 GB in the memory. Additionally, the amount of extra memory needed for training the model depends largely on the cho-

[10] https://github.com/facebookresearch/DPR

[11] https://github.com/IvaBojic/SleepQA

sen batch size. Once a single BERT model have loaded into the GPU memory, the available space left can be less than 1GB, as a result, the maximum batch size that this study can use is limited to 3, whereas the original paper obtains a batch size of 16.

Augmenting the SleepQA training set took 4 hours to paraphrase the 4000 training samples one time. As a result, the process of producing the SleepQA 3x dataset for this study requires 8 hours to complete.

When training the retrieval model, using the original SleepQA dataset which contains 4000 question-passage pairs, one epoch typically requires around 7 minutes. With the larger SleepQA 3x dataset that includes 12,000 question-passage pairs, a single epoch can take over 20 minutes. For traing the reader model, one epoch takes about 5 minutes. This study runs 30 epochs for each experiment, same as in the original paper.

4 Results

4.1 Retrieval - Reproduction study

This study chose PubMedBERT for experiment as it is the best-performing fine-tuned retriever of the original paper. By employing the checkpoint of fine-tuned PubMedBERT provided by the authors, the study is able to produce the recall@1 score of the reported value.

However, when training and fine-tuning PubMedBERT from stratch, despite using the same DPR framework and codebase as the original paper did, this study's findings did not yield comparable recall@1 score to that of the original paper (0.35 vs 0.42), although it matches the recall@1 score of the fine-tuned general-domain BERT base model reported in the original paper (See table 4).

The study conducted various experiments using different combinations of batch sizes (2 and 3) and learning rates (1e-05, 1e-06 and 2e-6), and it was observed that the best result of 0.35 was obtained using a batch size of 3 and a learning rate of 2e-6.

4.1.1 Retrieval - Ablation study for DPR BERT model's effectiveness without fine-tuning on SleepQA

To assess the extent of improvement that can be achieved from fine-tuning BERT models with the SleepQA train set and understand the value of SleepQA dataset, this study finds it necessary to compare an off-the-shelf BERT model with the authors' fine-tuned model on SleepQA. As this comparison was not performed in the original study, this work conducts it and tests on the SleepQA test set. For the baseline model, this study uses DPR BERT-based retriever (consisting of a question encoder [12] and a context encoder[13]) trained on Natural Questions (NQ) dataset that has 58,880 training samples (Karpukhin et al., 2020).

The outcome shown in table 4 demonstrates that PubMedBERT, when fine-tuned with SleepQA, outperforms BERT NQ. This upholds the importance of utilizing domain-specific datasets for domain-specific tasks, specifically, the value of SleepQA dataset to sleep health domain, despite their significantly smaller size (4,000) when compared to open-domain datasets (58,880).

4.1.2 Retrieval - Ablation study with SleepQA 3x dataset

As this study did not attain a recall@1 score comparable to that presented in the original paper after fine-tuning PubMedBERT with SleepQA, it advanced to expand the dataset size and examine the benefits of a larger dataset within the same domain.

The findings in table 4 indicate that augmenting the SleepQA dataset by paraphrasing to three times its original size, or SleepQA3x, does not lead to an improvement in the recall@1 score. Within 30 epochs, the best recall@1 scores obtained from using SleepQA and SleepQA 3x are the same, except for SleepQA3x achieving the highest score at epoch 15 while SleepQA converges at a slower pace at epoch 23 in the experiment.

Model	recall@1
Fine-tuned BERT (SleepQA)*	0.35
Fine-tuned PubMedBERT (SleepQA)*	**0.42**
DPR BERT (NQ)†	0.18
Fine-tuned PubMedBERT (SleepQA)†	0.35
Fine-tuned PubMedBERT (SleepQA3x)†	0.35

Table 4: Retrieval results. Symbol *: The scores are directly taken from the orginial paper (Bojic et al., 2022); Symbol †: The scores are obtained in this work.

4.2 Reader - Reproduction Study

To verify reader performance, this study experimented with BioBERT BioASQ model, which was

[12] https://huggingface.co/facebook/dpr-question_encoder-single-nq-base

[13] https://huggingface.co/facebook/dpr-ctx_encoder-single-nq-base

identified as the highest-performing reader in the original paper. Like the original paper, this study trained reader models independently from the retrieval models, using the question and its exact passage ("oracle") as inputs for the reader.

During the process of fine-training the model from scratch, this study was able to attain EM and F1 scores that matched those in the original paper (See Table 5), and this result was achieved at as soon as the second epoch. Checkpoint beyond the the second epoch did not lead to better improvement. This finding confirms both the reader performance reported in the original paper and the fact that fine-tuning with SleepQA can generate better outcomes compared to the baseline model (BERT SQuAD2).

Model	EM (oracle)	F1 (oracle)
BERT SQuAD2*	0.50	0.64
Fine-tuned BioBERT BioASQ*	0.61	0.73
Fine-tuned BioBERT BioASQ†	0.61	0.72

Table 5: Reader results. Symbol *: The scores are directly taken from the orginial paper (Bojic et al., 2022); Symbol †: The scores are obtained in this work.

5 Discussion

5.1 Retriever reproduction performance

This reproduction study attempted to train the PubMedBERT model from scratch for retrieval, but it did not yield the same outcome as the original paper. The study used the same DPR codebase and evaluation metrics as the original paper, but with a much smaller batch size and a lower learning rate (adjusted due to the smaller batch size). A batch size of 3 is the maximum size achieved with the memory constraint, though it is an uncommon size, it was observed that a batch size of 3 performed better than a batch size of 2 in the experiment. The use of a small batch size could potentially lead to underperformance of the retrieval task in the study, as the DPR framework's learning objective is based on contrastive learning.

The DPR framework utilizes in-batch negative technique (Karpukhin et al., 2020), where for each question in each batch, its relevant passage is a positive sample and the relevant passages of other questions within the same batch are used as negative samples. Let N be the number of questions in a batch and each one is associated with a relevant passage, this leads to B training instances in each batch, where there are $N - 1$ negative passages for each question.

Therefore, given a batch size of 3 in this reproduction, the number of negative pairs per question was only 2, whereas the original paper used a batch size of 16 and had 15 negative pairs per question.

As the contrastive learning is to optimize the loss function as the negative log likelihood of the positive passage (Karpukhin et al., 2020), inadequate negative passages can result in sub-optimal training performance.

5.2 Reader reproduction performance

In contrast to the retrieval model, this study was able to achieve the similar level of performance as the original paper after only 2 epochs of fine-tuning BioBERT BioASQ as a reader from scratch, despite using a smaller batch size of 3 compared to the original paper's batch size of 16.

The success of reader reproduction can be attributed to the fact that the reader model does not involve contrastive learning. Unlike retriever, the reader is built upon the question-answering BERT mechanism (Devlin et al., 2019) by training start and end token classifiers with softmax activation. The two classifiers identify the start and end words with the highest probability over all the words in the relevant passage and then extract the answer span from the identified start and end words. Therefore, the batch size has less of an impact on training performance.

5.3 The effectiveness of larger dataset SleepQA3x

The study aimed to overcome the limitations of a small batch size and enhance the retrieval performance by generating a more extensive dataset. However, the findings indicate that the augmented dataset did not contribute to improving the results.

The reason data augmentation through paraphrasing is not effective can be that paraphrasing is done through using synonym, altering word order, modifying sentence structure while preserving the original meaning. The objective of DPR retrieval is to produce dense embeddings that can better capture the semantics of sentences and enhance the retrieval of relevant passages. Paraphrasing that preserves the original meaning fails to diversify or augment the semantic information already present in the original training set. Consequently, creating a larger training set by paraphrasing existing samples is ineffective in train a better retrieval.

This experiment on paraphrasing also examines the authors' suggestion of employing back translation for data augmentation in future work. Since back translation is essentially the same as paraphrasing in terms of preserving semantics, the use of back translation might reduce the similarity between questions and answers in the original dataset, thus making fine-tuned BERT model's performance less distinct in comparison to BM25, however, it would not be beneficial in enhancing the performance of fine-tuned retrievers, as demonstrated in this study.

5.4 What was easy

In this reproduction study, working with the dataset was relatively easy as it has been extensively collected, carefully curated, and neatly formatted for model training. The authors made it readily available in their GitHub repository, which eliminated the need for data preprocessing.

Also, the utilization of the DPR framework built on Hydra (Yadan, 2019) for fine-tuning the BERT models in the original paper makes reproducing this study relatively simple. Following the authors' instructions, one needs to only modify the configuration files to set the retrieval and reader models, adjust the configuration and hyperparameters.

5.5 What was difficult

While the DPR framework is available for use, its structure and codebase are of such complexity that require advanced programming knowledge to comprehend fully. It also took additional time to understand the framework completely and implement it properly.

Additionally, replicating the work of the original paper, which involves fine-tuning BERT-based language models, requires significant computational resources, including powerful GPUs, and a substantial amount of memory. While a GPU was utilized, fine-tuning took 3.5 hours for a retriever and 2.5 hours for a reader, for 30 epochs to align with the original paper. With over 100 million parameters, the BERT models require a large amount of memory to reach the batch size used in the original paper. Due to the memory limitation in this study, reproducing the retrieval results presented in the original paper was challenging. To attempt to replicate the results, this study had to run various experiments, including trying different combinations of batch size and learning rate and creating a new larger dataset, which accounted for the majority time of this study.

5.6 Recommendations

A more diverse and extensive dataset would be necessary to improve the accuracy of the retrieval model. In the DPR paper (Karpukhin et al., 2020), it was demonstrated that dense retrievers trained on larger datasets such as NQ, TriviaQA, and SQuAD v1.1, outperformed BM25 in terms of retrieval accuracy, given that the authors' fine-tuned BERT models did not surpass the performance of BM25 and their highest recall@1 score of 0.42 is still suboptimal to make an effective retriever of specific domain, a larger dataset would be beneficial in improving the retrieval model's accuracy.

For reproducibility of this paper, it is essential to understand the DPR methodology (Karpukhin et al., 2020) and become familiar with its framework and codebase. This would aid in comprehending the fine-tuning process and the configuration files used to set the retrieval and reader models and their hyperparameters.

Last but not least, having adequate computational resources, including computing power and a substantial amount of memory, is critical to ensure reproducibility. Fine-tuning BERT-based language models requires powerful GPUs or TPUs and a large amount of memory, which can significantly impact the experimental results. Therefore, it is necessary to ensure that the computational resources used for replication are comparable to those utilized in the original study.

References

Iva Bojic, Qi Chwen Ong, Megh Thakkar, Esha Kamran, Irving Yu Le Shua, Jaime Rei Ern Pang, Jessica Chen, Vaaruni Nayak, Shafiq Joty, and Josip Car. 2022. Sleepqa: A health coaching dataset on sleep for extractive question answering. In *Proceedings of the 2nd Machine Learning for Health symposium*, volume 193 of *Proceedings of Machine Learning Research*, pages 199–217. PMLR.

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

R. Hadsell, S. Chopra, and Y. LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Omry Yadan. 2019. Hydra - a framework for elegantly configuring complex applications. Github.

Juan Yang, Brent A Bauer, Stephanie A Lindeen, Adam I Perlman, Kasey R Boehmer, Manisha Salinas, Susanne M Cutshall, et al. 2020. Current trends in health coaching for chronic conditions: A systematic review and meta-analysis of randomized controlled trials. *Medicine*, 99(30).

Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021. Retrieving and reading: A comprehensive survey on open-domain question answering.