Exploring Approaches to Answer Biomedical Questions: From Pre-processing to GPT-4

Notebook for the BioASQ Lab at CLEF 2023

Hyunjae Kim[1], Hyeon Hwang[1], Chaeeun Lee[1], Minju Seo[1], Wonjin Yoon[2,3,†] and Jaewoo Kang[1,4,*]

[1]Department of Computer Science and Engineering, Korea University, Seoul, 02841, Republic of Korea
[2]Computational Health Informatics Program, Boston Children's Hospital, Boston, MA, 02115, USA
[3]Harvard Medical School, Boston, MA, 02115, USA
[4]AIGEN Sciences, Seoul, 04778, Republic of Korea

Abstract

Biomedical question answering (QA) plays a crucial role in assisting researchers, healthcare professionals, and even patients in accessing and retrieving accurate and up-to-date information from the vast amount of biomedical knowledge available in literature. To enhance the efficiency of knowledge discovery and information retrieval, we investigate the efficacy of various pre-processing, model training, data augmentation, and ensemble methods and evaluate a range of advanced pre-trained models such as BioLinkBERT and GPT-4. Additionally, we explore data augmentation and ensemble methods to further improve system performance. In our participation in BioASQ Task 11b-Phase B, our systems achieved a top ranking in all four batches for the yes/no type of questions, in one out of four batches for factoid questions, and in two out of four batches for list-type questions.

Keywords

BioASQ 11b, BioLinkBERT, GPT-4, Data Augmentation, Ensemble

1. Introduction

Biomedical question answering (QA) is a pivotal tool, empowering researchers, healthcare professionals, and patients to access accurate, up-to-date information from the vast pool of biomedical knowledge in the literature. The BioASQ challenge [1] has actively fostered collaborative efforts across the scientific community to push the boundaries of cutting-edge biomedical QA research for over a decade. Yoon et al. [2] made a significant contribution to the biomedical QA field by utilizing the pre-trained model BioBERT [3], instead of traditional word embedding models, in their milestone study. This approach has paved the way for numerous other systems and approaches in this field.

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

[*] Corresponding author.

[†] WY was affiliated with Korea University during the early stage of this work.

hyunjae-kim@korea.ac.kr (H. Kim); hyeon-hwang@korea.ac.kr (H. Hwang); chaeeunlee1997@korea.ac.kr (C. Lee); minjuseo@korea.ac.kr (M. Seo); wonjin.yoon@childrens.harvard.edu (W. Yoon); kangj@korea.ac.kr (J. Kang)

CEUR Workshop Proceedings (CEUR-WS.org)

In this paper, we explore understudied or recently proposed pre-processing techniques, new pre-trained models, training objectives, data augmentation, and ensemble approaches in the BioASQ Task 11b [4]. We first revisit existing pre-processing methods to analyze what type of question each method is suitable for. We use BioLinkBERT [5] as a new embedding model in the BioASQ challenge and show that it outperforms existing language models such as BioBERT and PubMedBERT [6]. We adopt a sequence tagging approach [7, 8] to train list-type QA models, which is the first attempt in the challenge. In addition, we investigate the potential performance improvement in BioASQ by employing data augmentation techniques using SQuAD [9], a human-labeled factoid dataset consisting of Wikipedia documents, and LIQUID [10], an automated framework that generates list-type questions and corresponding answers from PubMed abstracts. We examine whether the utilization of an ensemble method can further optimize the performance. Finally, we evaluate the capability of the state-of-the-art pre-trained model, GPT-4 [11], on the list-type questions in a one-shot manner.

We selected the best combination of approaches from pre-processing to the use of GPT-4, through experiments on the BioASQ-10b dataset [12] for each question type. We participated in the BioASQ Task 11b-Phase B [13] to evaluate our systems on the official leaderboard. Our systems delivered remarkable performance across various question types, leading to impressive rankings. In the yes/no type, our systems achieved first place across all four batches, while in the factoid type, we attained the highest rank in one out of four batches. Additionally, we secured the highest rank in two out of four batches for the list type.

2. Task Description

In BioASQ 11B-phase b [13], models are required to provide answers to a given question, denoted as q. These answers are inferred from a collection of snippets, represented as $s_1, \ldots, s_J$, which are extracted from PubMed abstracts. The format of answers depends on the question types. For this year's competition, we focused only on the following three question types that require *exact* answers, excluding questions requiring *ideal* answers.

Yes/no. For this type of question, the model should answer "yes" or "no" to a question based on the given snippets. An example question of this type is: "*Is capmatinib effective for glioblastoma?*" Answering yes/no questions often requires considering multiple snippets collectively rather than relying on a single snippet alone.

Factoid. Factoid-type questions are mainly concerned with the confirmation or summarization of factual information and require a single concise answer (e.g., "*Which enzyme does Opicapone inhibit?*"). The ground-truth answers are usually, but not always, contained in the snippets, which distinguishes it from other QA tasks such as SQuAD, where answers always can be extracted in a given context [9].

List. The list type requires more than one answer to a single question. An example question is: "*What laboratory abnormalities are commonly seen in patients with COVID-19?*" Although list-type questions have received less attention in academic research compared to factoid-type questions [14], they are frequently encountered in practice, especially in the biomedical domain [8]. Similar to the factoid type, answers may or may not be extracted from the given snippets.

3. Methods

Our system comprises three models, each specifically designed for a particular question type. To identify the optimal choices for each question, we explore five factors to consider in effectively addressing different types of questions: pre-processing techniques (Section 3.1), QA model selection (Section 3.2), training objectives (Section 3.3), data augmentation approaches (Section 3.4), and the use of ensemble methods (Section 3.5).

3.1. Pre-processing

Given that the total length of all J snippets might exceed the input length limit of language models, it becomes necessary to select which snippets from the given set should be provided to the model as input. We examine two pre-processing techniques: the "single snippet" method and the "full snippet" method.

Single snippet. One straightforward approach is to treat each snippet as an individual instance, resulting in multiple question-snippet pairs as follows: $(q, s_1), \ldots, (q, s_J)$. For the factoid and list types, only snippets that contain at least one answer string are utilized as training instances. On the other hand, all snippets are used for training for the yes/no type. Although this approach has been commonly used in previous studies because of its simplicity, treating each snippet as a separate instance can be sub-optimal, particularly for the yes/no type. In answering yes/no questions, it is often necessary that multiple snippets are considered collectively and the information present in one snippet should be considered in conjunction with other pieces of information available in other snippets. Furthermore, this method disregards the opportunity to extract valuable information from other snippets that could potentially aid in predicting correct answers.

Full snippet. The full-snippet method addresses the limitations of the single-snippet approach by concatenating all snippets together to form a single comprehensive evidence context. In cases where the question-context pair exceeds the input length limitation imposed by the language model, the set of given snippets is partitioned into multiple contexts based on sentence boundaries. In other words, each context is created by concatenating the maximum number of snippets, ensuring that the length limit is not exceeded. Separate contexts created in this manner are then treated as individual instances.

3.2. QA Model

BioLinkBERT. BioLinkBERT [5] is a pre-trained language model trained on PubMed abstracts using a new pre-training objective called document relation prediction, where the model predicts whether two different segments are linked,[1] come from a single document, or are randomly selected from different documents. The model outperformed existing biomedical language models such as PubMedBERT in various downstream tasks. Especially the model achieved an accuracy of 94.8 on yes/no questions in a previous BioASQ challenge dataset [15]. Inspired by this result, we used a BioLinkBERT-large model as our backbone model. For a given question q and context c that consists of one or multiple snippets, the corresponding token representations are encoded as follows:

$$[\mathbf{h}_{[\texttt{CLS}]}, \mathbf{h}_{q_1}, \dots, \mathbf{h}_{q_T}, \mathbf{h}_{[\texttt{SEP}]}, \mathbf{h}_{c_1}, \dots, \mathbf{h}_{c_L}] = E(q, c), \tag{1}$$

where E is the BioLinkBERT encoder, $\mathbf{h}_{[\texttt{CLS}]} \in \mathbb{R}^d$ and $\mathbf{h}_{[\texttt{SEP}]} \in \mathbb{R}^d$ are the representations of special tokens [16], and T and L are the lengths of the question and context, respectively. The token representations are then fed into task-specific layers, which will be described in Section 3.3.

GPT-4. Recently, foundation models such as ChatGPT [17] and GPT-4 [11] have been utilized in various downstream applications. Notably, they have demonstrated comparable or even superior performance compared to supervised models without fine-tuning. These findings serve as compelling evidence that the models possess the capability to deliver accurate answers to questions even in specialized domains such as biomedicine. In this challenge, we selected Open AI's latest model, GPT-4, as our QA system.[2] Unlike BioLinkBERT, GPT-4 is a black-box model; thus, we cannot access the hidden representation to update the model, and we should query the model using instructions [20]. We used only a single labeled example to provide the model with more comprehensive information on the task and desired output format (see Table 1 for the input prompt we used).

3.3. Training Objective

Binary classification. For the yes/no type, the model is trained using a binary classification objective, where the final hidden representation of the `[CLS]` token, $\mathbf{h}_{[\texttt{CLS}]}$, is fed to a linear layer. The loss is defined as the sum of the negative log probabilities of the true answer class (i.e., yes or no).

Span prediction. For the factoid type, the token representations of the context are fed into two different linear layers that calculate logit values for the start and end positions of the answer span as follows: $z^{\text{start}}_{c_1}, \dots, z^{\text{start}}_{c_L} = [\mathbf{w}^{\top}_{\text{start}}\mathbf{h}_{c_1}, \dots, \mathbf{w}^{\top}_{\text{start}}\mathbf{h}_{c_L}], z^{\text{end}}_{c_1}, \dots, z^{\text{end}}_{c_L} = [\mathbf{w}^{\top}_{\text{end}}\mathbf{h}_{c_1}, \dots, \mathbf{w}^{\top}_{\text{end}}\mathbf{h}_{c_L}]$. These logits are used to calculate probability values for each token, indicating the likelihood of it being the start or end of the answer span. The loss is calculated by summing the negative log probabilities of the start and end positions of the ground-truth answer.

[1] Citation information were used.

[2] Note that we used GPT-4 only in the list type because the model did not outperform supervised models for the yes/no and factoid types in our initial experiments on BioASQ-10b. Please see concurrent works for results of GPT models in yes/no and factoid questions in the biomedical domain [18, 19].

Table 1
The designed prompt for GPT-4 to answer list-type questions. {*Test Question*} and {*Test Context*} are substituted by real questions and contexts in test batches.

Your task is to identify a list of answers to the question in the provided context.
To help you understand the task, here is an example:

Question:
Which acetylcholinesterase inhibitors are used for treatment of myasthenia gravis?

Context:
Pyridostigmine and neostygmine are acetylcholinesterase inhibitors that are used as
first-line therapy for symptomatic treatment of myasthenia gravis.

Answer:
neostigmine, pyridostigmine

Now, here is the actual question and context for you to find the appropriate list of answers.

Question:
{*Test Question*}

Context:
{*Test Context*}

Answer:

Sequence tagging. One conventional approach to solving list-type QA involves considering the top answer predictions of a single-span QA model that surpasses a pre-defined threshold as final predictions. Recent studies [7, 8] have proposed an alternative approach, treating list QA as a sequence tagging problem, where the model classifies each context token into B (beginning), I (inside), or O (outside) tags, similar to named entity recognition. This approach showed better performance in list QA than existing single-span QA models in a range of general and biomedical QA datasets [14, 10]. Inspired by these results, we adopted this approach to train our list-type QA model.

3.4. Data Augmentation

We explored two data augmentation approaches to enhance performance in the factoid and list types. [3]

SQuAD. For the factoid type, we followed previous studies that leveraged a large-scale single-span dataset SQuAD [2, 22, 23, 24]. While the SQuAD dataset is not specifically designed for the

[3] Due to the high performance of our models in initial experiments, we did not extensively investigate a data augmentation approach for the yes/no question type. However, it would be interesting to investigate the impact and transferability of existing yes/no QA datasets such as PubMedQA [21] in future research.

Table 2
Statistics of the BioASQ-10b (the training and validation sets) and BioASQ-11b (the test batches) datasets.

Type	Number of Questions					
	Training	Validation	Batch 1	Batch 2	Batch 3	Batch 4
Yes/no	1,148	124	24	24	24	14
Factoid	1,252	166	19	22	26	31
List	816	85	12	12	18	24

Table 3
Model components for each question type. We selected the best options using a validation process on BioASQ-10b.

Type	Pre-processing	QA Model	Training Objective	Data Augment.	Ensemble
Yes/no	Full snippet	BioLinkBERT [5]	Binary classification	✗	✓
Factoid	Single snippet	BioLinkBERT [5]	Span prediction	SQuAD [9]	✓
List	Full snippet	BioLinkBERT [5]	Sequence tagging	LIQUID [10]	✓
		GPT-4 [11]	✗	✗	✗

biomedical domain, it shares a fundamental similarity with the factoid-type QA in BioASQ. Both datasets aim to find accurate answers to factual questions within a provided text. We initially pre-trained our models using SQuAD and subsequently fine-tuned them using the BioASQ data.

LIQUID. A recent study [10] proposed a data generation model for list QA, called LIQUID, and made the 140k question-answer pairs produced by the model publicly available.[4] We utilized this synthetic data to pre-train our models, and subsequently, we fine-tuned the models using the BioASQ data.

3.5. Ensemble

We used different ensemble techniques for the yes/no, factoid, and list types, respectively.

Yes/No. We employed majority voting, where predictions from each individual model were aggregated, and the final prediction was determined by the majority prediction.

Factoid. We use a probability-based ensemble method for factoid-type questions. In this approach, we calculate the sum of probability values for each of the top 20 answers predicted

[4] https://github.com/dmis-lab/LIQUID

Table 4
Performance (macro F1) on BioASQ-11b in the yes/no type. Numbers in parentheses indicate the number of single models constituting the ensemble model.

Batch	Rank	System	Macro F1	Description
Batch 1	1	DMIS-KU-5	0.9515	Single
	7	DMIS-KU-1	0.8571	Ensemble (20)
	7	DMIS-KU-2	0.8571	Ensemble (10)
	7	DMIS-KU-3	0.8571	Ensemble (10)
	13	DMIS-KU-4	0.8545	Single
Batch 2	1	DMIS-KU-4	1.0000	Single
	4	DMIS-KU-1	0.9577	Ensemble (20)
	4	DMIS-KU-2	0.9577	Ensemble (10)
	4	DMIS-KU-3	0.9577	Ensemble (10)
	10	DMIS-KU-5	0.9143	Single
Batch 3	1	DMIS-KU-4	1.0000	Single
	3	DMIS-KU-1	0.9545	Ensemble (20)
	5	DMIS-KU-5	0.9111	Single
	11	DMIS-KU-2	0.8693	Ensemble (10)
	12	DMIS-KU-3	0.8634	Ensemble (10)
Batch 4	1	DMIS-KU-1	1.0000	Ensemble (20)
	7	DMIS-KU-2	0.9181	Ensemble (10)
	7	DMIS-KU-5	0.9181	Single
	15	DMIS-KU-3	0.9048	Ensemble (10)
	17	DMIS-KU-4	0.8250	Single

by individual models. The top five predictions with the highest summed probabilities are then selected as the final answers.

List. We counted the number of answers predicted by single models to a given question based on their string form. For each answer, we calculated an ensemble score as the proportion of how many models out of the total number of models predicted the answer. For instance, suppose that model A, model B, and model C predict {"leprosy," "cirrhosis," "cholera"}, {"leprosy," "COVID-19"}, and {"cirrhosis"}, respectively, then the ensemble scores of each prediction as follows: leprosy (2/3), cirrhosis (2/3), cholera (1/3), and COVID-19 (1/3). If the score is higher than the threshold, we included the predicted answer in the final answer set; otherwise, we excluded it. We searched for the best threshold value using the BioASQ 10b dataset.

4. Experimental Setups

4.1. Dataset

We used the training and test sets of the BioASQ-10b dataset [12] as our training and validation set, respectively. Systems were evaluated on BioASQ-11b [25], which was newly proposed for

Table 5
Performance (mean reciprocal rank, i.e., MRR) on BioASQ-11b in the factoid type. Numbers in parentheses indicate the number of single models constituting the ensemble model.

Batch	Rank	System	MRR	Description
Batch 1	4	DMIS-KU-1	0.5526	Ensemble (10)
	4	DMIS-KU-2	0.5526	Ensemble (10)
	4	DMIS-KU-5	0.5526	Ensemble (10)
	7	DMIS-KU-4	0.5439	Ensemble (20)
	9	DMIS-KU-3	0.5088	Ensemble (10)
Batch 2	8	DMIS-KU-1	0.4773	Ensemble (15)
	11	DMIS-KU-4	0.4697	Ensemble (10)
	13	DMIS-KU-3	0.4621	Ensemble (15)
	13	DMIS-KU-5	0.4621	Ensemble (10)
	15	DMIS-KU-2	0.4561	Ensemble (20)
Batch 3	2	DMIS-KU-1	0.5154	Single
	4	DMIS-KU-2	0.5077	Ensemble (2)
	6	DMIS-KU-3	0.4981	Single
	13	DMIS-KU-5	0.4647	Single
	16	DMIS-KU-4	0.4500	Ensemble (2)
Batch 4	1	DMIS-KU-1	0.7323	Ensemble (4)
	2	DMIS-KU-2	0.7108	Ensemble (4)
	3	DMIS-KU-3	0.6882	Single
	4	DMIS-KU-4	0.6570	Single
	5	DMIS-KU-5	0.6473	Single

the 2023 challenge. The statistics of the datasets are listed in Table 2.

4.2. Our Systems

We selected our final systems through a validation process among various combinations of methods (see Section 5.2 for detailed validation results). Table 3 presents the optimal selections for the "single" model for each question type. We searched for the best checkpoints of single models by measuring performance on the validation set every epoch. Ensemble models consisted of different single models that were randomly initialized and then selected through the validation process.

5. Results

5.1. Official Evaluation on BioASQ-11b

Tables 4, 5, and 6 show that our best models achieved top scores in many batches. Especially, in the yes/no type, our models achieved the highest scores across all batches, to with full-snippet method contribute significantly.

Table 6
Performance (F-measure) on BioASQ-11b in the list type. Numbers in parentheses indicate the number of single models constituting the ensemble model. Please refer to Section 3.5 for a description of threshold values.

Batch	Rank	System	F-Measure	Description
Batch 1	1	DMIS-KU-3	0.7027	Ensemble (20), threshold: 0.6
	1	DMIS-KU-5	0.7027	Ensemble (20), threshold: 0.7
	3	DMIS-KU-4	0.6992	Ensemble (10), threshold: 0.7
	4	DMIS-KU-2	0.6965	Ensemble (10), threshold: 0.6
	5	DMIS-KU-1	0.6937	Ensemble (10), threshold: 0.6
Batch 2	6	DMIS-KU-3	0.3178	Ensemble (15), threshold: 0.75
	8	DMIS-KU-2	0.3087	Ensemble (15), threshold: 0.75
	9	DMIS-KU-1	0.3080	Ensemble (10), threshold: 0.6
	10	DMIS-KU-5	0.3022	Ensemble (20), threshold: 0.4
	13	DMIS-KU-4	0.2871	Ensemble (20), threshold: 0.4
Batch 3	4	DMIS-KU-5	0.5477	Ensemble (15), threshold: 0.75
	5	DMIS-KU-4	0.5466	Ensemble (15), threshold: 0.75
	6	DMIS-KU-3	0.5454	Ensemble (20), threshold: 0.7
	7	DMIS-KU-2	0.5437	Ensemble (20), threshold: 0.6
	8	DMIS-KU-1	0.5341	Ensemble (10), threshold: 0.6
Batch 4	1	DMIS-KU-1	0.7440	GPT-4
	4	DMIS-KU-2	0.6806	Ensemble (20), threshold: 0.6
	5	DMIS-KU-5	0.6787	Ensemble (15), threshold: 0.75
	6	DMIS-KU-3	0.6752	Ensemble (20), threshold: 0.7
	7	DMIS-KU-4	0.6747	Ensemble (15), threshold: 0.75

In the factoid type, we achieved the highest score in the last batch. Our factoid QA models basically used a similar model structure and training method, but their performance and rankings were very different from batch to batch. This is because we continuously searched for best single models by randomly initializing them, making us to obtain better single models in batches 3 and 4. In addition, we found that the performance of ensemble models depended on the individual performance of single models rather than the quantity of single models. For instance, by ensembling a small number of high-performing models, we were able to achieve second and first place in batches 3 and 4, respectively.

In the list type, we achieved first place in two batches using supervised model and GPT-4, respectively. For the supervised model, the full-snippet method, data augmentation using LIQUID, and ensemble were all effective to improve the performance (see Section 5.2 for more results). GPT-4 outperformed our supervised models in batch 4 and achieved the best performance. This is very surprising because our supervised models were ensemble models of several single models trained using thousands of human-labeled BioASQ data and 140k artificial QA data, while GPT-4 used only a single question-answer pair.

Table 7
Ablation study of pre-processing methods. See Section 3.1 for detailed descriptions of the single-snippet and full-snippet approaches.

Method	Yes/no	Factoid	List
Single snippet	0.9347	**0.5132**	0.4773
Full snippet	**0.9815**	0.4762	**0.5373**

Table 8
Comparision of pre-trained language models. Note that the performance was measured using macro F1 and mean reciprocal rank for the yes/no and factoid types, respectively.

Model	Vocabulary	Model Size	Yes/no	Factoid
BioBERT-base	Wiki+Books	110M	0.8091	0.4734
PubMedBERT-base	PubMed	110M	0.9630	0.4815
BioLinkBERT-base	PubMed	110M	0.9634	0.4840
BioLinkBERT-large	PubMed	340M	**0.9837**	**0.5132**

Table 9
Ablation study of data augmentation methods. Note that SQuAD and the synthetic dataset generated by LIQUID [10] were used as augmenting data for the factoid and list types, respectively.

Method	Factoid	List
BioASQ	0.5132	0.5373
+ Augment.	**0.5294** (+ 0.0162)	**0.5731** (+ 0.0358)

5.2. Ablation Study on BioASQ-10b

Effect of pre-processing. Table 7 shows that the effect of the pre-processing method varied depending on the type of question. In the case of yes/no and list question types, the full-snippet approach outperformed the single-snippet method. This is because both question types require a comprehensive understanding of the context to provide accurate answers. However, for the factoid question type, the single-snippet method was found to be more suitable. We speculate that the single-snippet method was effective because most factoid questions can be answered with only the surrounding context of the answer without much additional context.

Language model selection. To find the best-performing encoder on the BioASQ data, we tested several variants of common pre-trained language models in the biomedical domain: BioBERT [3], PubMedBERT [6], and BioLinkBERT [5]. As shown in Table 8, BioLinkBERT was slightly better than PubMedBERT with the same size (110M parameters), and the BioLinkBERT-large model significantly outperformed the base-sized models.

Effect of data augmentation. Table 9 shows that augmenting the SQuAD data improves performance on the factoid questions, which is consistent with previous studies [2, 22, 23, 24].

Table 10
Validation performance of single and ensemble models for the three question types. "Best Single" indicates the highest performance among the single models that constitute the ensemble model. The performance is measured based on Macro F1 scores for the yes/no type, mean reciprocal rank for the factoid type, and F-measures for the list type.

Type	Batch	System	Best Single	Ensemble
Yes/no	4	DMIS-KU-1	0.9815	0.9908 (+ 0.0093)
Factoid	4	DMIS-KU-1	0.5522	0.5643 (+ 0.0121)
List	4	DMIS-KU-2	0.5964	0.6127 (+ 0.0163)

In addition, the LIQUID data significantly improved the model performance on the list type, which is consistent with Lee et al. [10].

Effect of ensemble. Table 10 shows validation results for the three question types, highlighting improvements in performance attained through ensembling of multiple models.

6. Conclusion

This study focused on conducting comprehensive experiments, encompassing a range of preprocessing techniques and the utilization of advanced models such as BioLinkBERT and GPT-4. In addition, we delved into the exploration of data augmentation and ensemble methods, further refining the performance of our QA system. Our models achieved high performance in BioASQ task 11b - phase B. We hope that our findings and analysis will contribute towards enhancing the performance of biomedical QA systems, ultimately maximizing knowledge discovery and information retrieval efficiency.

Acknowledgments

This research was supported by (1) National Research Foundation of Korea (NRF-2023R1A2C3004176), (2) the MSIT (Ministry of Science and ICT), Korea, under the ICT Creative Consilience program (IITP-2023-2020-0-01819) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation), and (3) a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (grant number: HR20C0021(3)).

References

[1] G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos, et al., An overview of the bioasq large-scale biomedical semantic indexing and question answering competition, BMC bioinformatics 16 (2015) 1–28.

[2] W. Yoon, J. Lee, D. Kim, M. Jeong, J. Kang, Pre-trained language model for biomedical question answering, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2019, pp. 727–740.

[3] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2020) 1234–1240. URL: https://academic.oup.com/bioinformatics/article-abstract/36/4/1234/5566506.

[4] A. Nentidis, G. Katsimpras, A. Krithara, S. Lima-López, E. Farré-Maduell, L. Gasco, M. Krallinger, G. Paliouras, Overview of BioASQ 2023: The eleventh BioASQ challenge on Large-Scale Biomedical Semantic Indexing and Question Answering, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023), 2023.

[5] M. Yasunaga, J. Leskovec, P. Liang, LinkBERT: Pretraining language models with document links, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 8003–8016. URL: https://aclanthology.org/2022.acl-long.551. doi:10.18653/v1/2022.acl-long.551.

[6] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ArXiv preprint abs/2007.15779 (2020). URL: https://arxiv.org/abs/2007.15779.

[7] E. Segal, A. Efrat, M. Shoham, A. Globerson, J. Berant, A simple and effective model for answering multi-span questions, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 3074–3080. URL: https://aclanthology.org/2020.emnlp-main.248. doi:10.18653/v1/2020.emnlp-main.248.

[8] W. Yoon, R. Jackson, A. Lagerberg, J. Kang, Sequence tagging for biomedical extractive question answering, Bioinformatics 38 (2022) 3794–3801.

[9] P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang, SQuAD: 100,000+ questions for machine comprehension of text, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 2383–2392. URL: https://aclanthology.org/D16-1264. doi:10.18653/v1/D16-1264.

[10] S. Lee, H. Kim, J. Kang, Liquid: A framework for list question answering dataset generation, arXiv preprint arXiv:2302.01691 (2023).

[11] OpenAI, Gpt-4 technical report, ArXiv preprint (2023). URL: https://arxiv.org/abs/2303.08774.

[12] A. Nentidis, G. Katsimpras, E. Vandorou, A. Krithara, A. Miranda-Escalada, L. Gasco, M. Krallinger, G. Paliouras, Overview of bioasq 2022: the tenth bioasq challenge on large-scale biomedical semantic indexing and question answering, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2022, pp. 337–361.

[13] A. Nentidis, G. Katsimpras, A. Krithara, G. Paliouras, Overview of BioASQ Tasks 11b and Synergy11 in CLEF2023, in: Working Notes of CLEF 2023 - Conference and Labs of the Evaluation Forum, 2023.

[14] H. Li, M. Tomko, M. Vasardani, T. Baldwin, MultiSpanQA: A dataset for multi-span question answering, in: Proceedings of the 2022 Conference of the North American

Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Seattle, United States, 2022, pp. 1250–1260. URL: https://aclanthology.org/2022.naacl-main.90. doi:10.18653/v1/2022.naacl-main.90.

[15] A. Nentidis, K. Bougiatiotis, A. Krithara, G. Paliouras, Results of the seventh edition of the bioasq challenge, in: Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part II, Springer, 2020, pp. 553–568.

[16] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[17] OpenAI, Chatgpt blog post (2022). URL: https://openai.com/blog/chatgpt.

[18] I. Jahan, M. T. R. Laskar, C. Peng, J. Huang, Evaluation of ChatGPT on biomedical tasks: A zero-shot comparison with fine-tuned generative transformers, in: The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 326–336. URL: https://aclanthology.org/2023.bionlp-1.30.

[19] S. Ateia, U. Kruschwitz, Is chatgpt a biomedical expert?–exploring the zero-shot performance of current gpt models in biomedical tasks, arXiv preprint arXiv:2306.16108 (2023).

[20] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[21] Q. Jin, B. Dhingra, Z. Liu, W. Cohen, X. Lu, PubMedQA: A dataset for biomedical research question answering, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 2567–2577. URL: https://aclanthology.org/D19-1259. doi:10.18653/v1/D19-1259.

[22] S. Alrowili, K. Vijay-Shanker, Exploring biomedical question answering with biom-transformers at bioasq10b challenge: Findings and techniques, CEUR Workshop Bologna, Italy, 2022.

[23] Z. KADDARI, T. BOUCHENTOUF, Larsa at bioasq 10b: classical and novel approaches for biomedical document retrieval and question answering (2022).

[24] H.-H. Ting, Y. Zhang, J.-C. Han, R. T.-H. Tsai, Ncu-iisr/as-gis: Using bertscore and snippet score to improve the performance of pretrained language model in bioasq 10b phase b (2022).

[25] A. Krithara, A. Nentidis, K. Bougiatiotis, G. Paliouras, BioASQ-QA: A manually curated corpus for Biomedical Question Answering, Scientific Data 10 (2023) 170.