Enhancing Biomedical Document Ranking with Domain Knowledge Incorporation in a Multi-Stage Retrieval Approach.

Maël Lesavourey, Gilles Hubert

Enhancing Biomedical Document Ranking with Domain Knowledge Incorporation in a Multi-Stage Retrieval Approach.

Notebook for the BioASQ Lab at CLEF 2024

Maël Lesavourey[1], Gilles Hubert[1]

[1]IRIT lab, 118 Route de Narbonne, F-31062 TOULOUSE CEDEX 9

Abstract

This article presents the results we obtained during BioASQ Task 12B Phase A on document ranking. Our strategy is based on a two-stage retrieval approach composed of a retriever and a reranker. The retriever is based on BM25 scoring and RM3 query expansion. The ranker is a BERT cross-encoder pre-trained on a biomedical corpus (BioLinkBERT). We study the impact of incorporating domain knowledge (MeSH) into this Pretrained Language Model and build a voting system to combine the insights from multiple models. Independently from the challenge, we also investigate a way to reduce the number of input tokens to bypass BERT limitation of 512 tokens for the input sequence.

Keywords

biomedical document ranking, information retrieval, thesaurus-based knowledge, BERT cross-encoder, multi-stage retrieval

1. Introduction

Passage and document rankings are very important tasks in information retrieval (IR) systems as they facilitate users' navigation through different sources. With the recent breakthrough of generative artificial intelligence (AI), they are also used in Retrieval Augmented Generation (RAG) [1] workflow to help generative systems produce more precise answers to a given query.

In the academic field, the rise of open science and online information access multiply the amount of knowledge available. The drawback is that it is becoming harder to find a specific and precise piece of information. For this reason the BioASQ[1] initiative [2] proposes an annual evaluation campaign to solve several tasks of biomedical IR. Specifically, TaskB-PhaseA [3] of the challenge focuses on document retrieval and text snippets extraction.

In recent years, pre-trained language models (PLMs) [4, 5] have achieved state-of-the-art results on various natural language processing (NLP) tasks due to their ability to learn the semantic of various texts. However, the performance of such models tends to drop when they are applied to corpora from specific domains like biomedicine. Indeed, biomedical literature contains special features that exacerbate the semantic gap between general information and biomedical knowledge. One may cite as examples the polysemy of biomedical terms (tumor, neoplasm, cancer referring to the same concept) and the complex lexical structures (abbreviations, formulas, proper names, etc). One way to solve this problem is to use language models (LMs) pre-trained on biomedical corpora [6, 7, 8] but several works show that it is not enough to capture semantic relationships between terms in a document.

Following our work during last year's evaluation campaign [9], we investigated the impact of incorporating biomedical knowledge into PLMs trained on domain specific texts. More precisely, we propose to modify the input sequence of a BERT-based model [4] by tagging its biomedical terms in

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

mael.lesavourey@irit.fr (M. Lesavourey); gilles.hubert@irit.fr (G. Hubert)

https://www.irit.fr/~Gilles.Hubert/ (G. Hubert)

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

order to direct the attention mechanism towards them. We combine this approach with a voting system to take advantage of several models outputs at the same time.

2. Method

2.1. Task Description

We participated in the TaskB-PhaseA on document retrieval. The campaign was divided in 4 batches of 85 queries each. Participating systems were expected to return, for each query, a ranked list of at most 10 relevant articles. These articles needed to be retrieved from PubMed Annual Baseline for 2024 (37 million citations). The evaluation metric used to build the leaderboard is the Mean Average Precision (MAP) due to its capability to take into account the order of the submitted items while being less restrictive than the traditional Precision score.

2.2. Systems Description

The strategy we used to answer this task is a multi-stage retrieval approach [10] which aims at dividing the document ranking into different phases. We built a two-stage pipeline with a retriever and a reranker. Traditionally, a retriever is based on bag of words (BoW) representations [11] and aims at creating a candidate list of hundreds of documents from the whole corpus (several millions). It is often less effective but also have lower computational costs. On another hand, the goal of the reranker is to build the final list of a dozens of documents from the list generated by the retriever. State-of-the-art models for this task are PLMs based on transformers [12] architecture which greatly enhance results but also increase the computational cost. Finally, we explore a way of incorporating biomedical knowledge into PLMs using the Medical Subjects Headings[2] (MeSH) thesaurus. This workflow is illustrated in Figure 1.

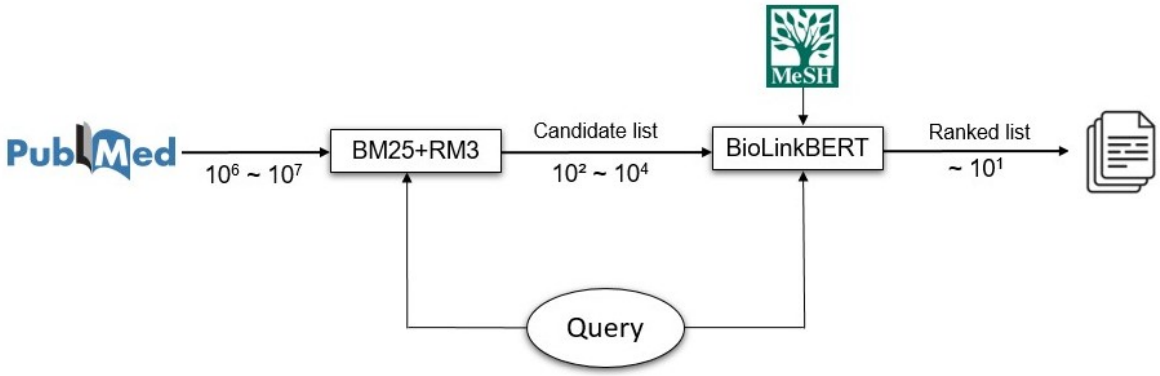

Figure 1: Pipeline of our retrieve then rerank approach.

In order to generate the candidate lists, we created our own index with PubMed articles and queried it using BM25 and RM3 [13]. BM25 is a term weighting model for evaluating document relevance, while RM3 uses pseudo-relevance feedback to improve search performance by incorporating additional information from relevant documents (query expansion). We used Pyserini [14], a Python toolkit for reproducing IR tasks, to create the index and the retriever.

Our base model for the reranking part is a BERT cross-encoder pre-trained on biomedical publications. Following the results of the BLURB[3] [15] leaderboard and due to computational limitations, we worked with the base version of BioLinkBERT [7]. We built a cross-encoder architecture trained with a pairwise approach. It takes as input the query and candidate document with the following sequence:

[2]https://www.nlm.nih.gov/mesh/meshhome.html
[3]https://microsoft.github.io/BLURB/

$[CLS]q_1, ..., q_n[SEP]d_1, ..., d_m[SEP]$. $(q_i)_{i\in[1,n]}$ and $(d_i)_{i\in[1,m]}$ being respectively the terms of the query and the document. The fine-tuning is done by computing a relevance score between the query and 4 documents (2 positive and 2 negative ones). The objective is to predict if each one is relevant or not using the $[CLS]$ token embedding, which is the pooled output of the model and represents the whole input. This embedding is passed through a single linear layer that produces the classification scores (logits). Applying the $SoftMax$ function on these scores allows to create a probability distribution on our classes (relevant and non-relevant). During inference, documents from a candidate list are ranked with their probability of being relevant for a given query.

To enhance the learning of biomedical entities and their semantic relations in the text, we propose to incorporate biomedical knowledge into the cross-encoder by modifying its input sequence. Our intuition is that a term referenced in a knowledge base will bring a greater piece of information than other words in the text. We extracted this knowledge from the MeSH thesaurus which is a controlled vocabulary maintained by the National Library of Medicine[4] (NLM) and used to index every citation in MEDLINE/PubMed. To incorporate MeSH terms into our PLM, we propose to tag the input sequence with a unique special token " # ". We built a vocabulary with all Main headings, Qualifiers and Suplementary Records and their corresponding Entry Terms of the MeSH thesaurus. We detected those terms in the query with an exact match between 1,2,3-grams in the query and the vocabulary. Then, we added the special token before and after each detected term in the query and their corresponding exact-matches and synonyms in the document. The idea is to guide the attention mechanism towards biomedical terms using this soft-matching of biomedical items. An example of the input sequence tagging is given in Figure 2.

Tagged query:
Is #galcanezumab# effective for #treatment# of #migraine#?

Tagged document:
100% Response Rate to #Galcanezumab# in Patients With Episodic #Migraine#: A Post Hoc Analysis of the Results From Phase 3, Randomized, Double-Blind, Placebo-Controlled EVOLVE-1 and EVOLVE-2 Studies. To characterize adult patients with episodic #migraine# who achieved 100% response to #galcanezumab# #treatment#. #Galcanezumab# is a humanized monoclonal antibody that selectively binds to the calcitonin gene-related peptide (CGRP) and has demonstrated efficacy in reducing #migraine headache# days (MHD) in patients with episodic and chronic #migraine#.

Figure 2: Example of the marking strategy on a query and the first sentences of one of its most relevant documents.

Finally, we learnt from last year experiments that systems' performances are inconsistent depending on the batches. To level them off, we built a voting system to mix the results of all our models along the batches. For a given query, we assigned a score to each of the 10 articles returned by each model. The voting system returns a new list of 10 items sorted along the scores obtained.

2.3. Additional Study

One of the main limitation of BERT-based model is the length of the input sequence [16]. Indeed such systems only take as input up to 512 tokens which represents even less words. This limit is easily exceeded when computing scientific publications even when considering only their title and abstract. In our systems described in section 2.2, we chose to simply truncate the input sequence and kept the first tokens until we reach the limitation. This method still provides competitive results in various tasks but lead to a loss of information as it deliberately ignores a part of the text. To overcome this problem, we fine-tuned another BioLinkBERT cross-encoder to compute the similarity between a sentence and a

[4] https://www.nlm.nih.gov/

query. This allows us to rank the sentences of a document by order of relevance to a given query. Thus, we reduced the input length by selecting the most relevant sentences without exceeding 512 tokens.

3. Experimental Settings

3.1. Data

To build our index we used the PubMed Annual Baseline[5] for 2024 from which we removed all articles without available abstracts. We concatenated the title and abstract of each citation to obtain a document. We also used the datasets released by the BioASQ[6] team [17] which contain all queries and their gold standards (relevant documents) from past editions.

We created a training set for the BERT cross-encoder. For each query we retrieved 4000 articles using BM25+RM3. The first 20 articles that were not in the gold standard were chosen as hard-negative samples while all the others were chosen as positive ones. We selected negative samples close to the query in terms of BoW embeddings as it enhances inference of BERT-based model[18].

In our last year experiments, we only extracted Main Headings and their corresponding Entry Terms from MeSH thesaurus. This year we also added Qualifiers and Supplementary Records in order to detect more biomedical and chemical concepts in the text. We used the 2024 release of MeSH[7] to apply the tagging strategy during both training and inference.

For inference, we generated a candidate list of 500 articles to be reranked by 4 PLMs as described in section 3.2.

3.2. Systems Settings

We used the same retriever, with the same parameters for every submission ($k1 = 0.6$, $b = 0.6$, $fb_terms = 16$, $fb_docs = 2$, $o_qw = 0.9$) [14].

In Table 1, we wrote down the settings of the 4 PLMs we trained and their corresponding submission name. BioASQ10 and BioASQ11 are 2 datasets provided by BioASQ teams containing all queries and their gold standards from the first year to respectively the 10th and 11th challenges. Note that for each batch, we selected two positive and two negative publications as described in 3.1. Fine-tuning was done using BertForSequenceClassification[8] with 2 classes (relevant or irrelevant) and we used the go-to loss function from the HuggingFace tool.

The voting system is inspired by the slate-vote as each voter (here the reranking systems) has to rank a list of candidates. We assigned a score for every document in the four generated lists of each query. Here are the scores assigned to the top 10 articles sorted by decreasing order of relevance: $[25, 19, 15, 12, 10, 8, 6, 5, 4, 4]$. A document is assigned a score of 0 for each list it does not appear in. Then, for every publication, we summed its 4 scores and used it to sort the documents and create a final list of the 10 most relevant papers. Note that a publication always in the bottom 4 of the ranked lists has a lower final score than a document ranked 1st by one system and not appearing in the other slates. This allows to reward a system that notices a strong similarity between the query and a document but would be left aside by other systems.

The PLM used to reduce the input sequence length has the same settings than "IRIS1" except for the data. We used as positive samples the "ideal answers" of each query (handwritten answers provided by BioASQ) and as negative samples sentences from the 50 first documents (except gold standard) retrieved by BM25+RM3. We used this model in two different ways:

- to reduce input length during inference and before applying "IRIS1". We will refer to this model as "IRIS1-R".

[5] https://ftp.ncbi.nlm.nih.gov/pubmed/baseline/
[6] http://participants-area.bioasq.org/datasets/
[7] https://www.nlm.nih.gov/databases/download/mesh.html
[8] https://huggingface.co/docs/transformers/v4.42.0/en/model_doc/bert#transformers.BertForSequenceClassification/

Table 1
Reranking systems parameters

Submission name	Model description	Tags	Training Corpus	Training parameters
IRIS1	BioLinkBERT-base	N/A	BioASQ10	LR=e^{-4}, Epochs=5, Batch=128×16
IRIS2	BioLinkBERT-base	N/A	BioASQ11	LR=e^{-4}, Epochs=5, Batch=128×16
IRIS3	Voting System	# or N/A	N/A	N/A
IRIS4	BioLinkBERT-base	#	BioASQ10	LR=e^{-4}, Epochs=5, Batch=128×16
IRIS5	BioLinkBERT-base	N/A	BioASQ10	LR=e^{-5}, Epochs=5, Batch=128×16

- during training and inference with the same settings as "IRIS1", i.e., we train the same model with documents having their input modified. We will refer to this model as "IRIS-R".

4. Results

In this section, we present the results obtained during batches 1, 2, 3, and 4 of the Task 12B Phase A on document ranking. We officially participated in batches 1, 2, and 3 of this year evaluation campaign with five models that are reported in Table 2 in which we also include unofficial results on batch 4. Additional results for systems including input length reduction are written down in Table 3.

Table 2
Results for Task 12B Phase A

Batch	Model	MAP	Recall	System Rank	Team Rank
1	IRIS1	0.1357	0.2339	13/40	3
-	IRIS2	0.1318	*0.2413*	14/40	3
-	IRIS3	**0.1471**	0.2404	11/40	3
-	IRIS4	0.1467	0.2389	12/40	3
-	IRIS5	0.1132	0.2365	18/40	4
2	IRIS1	0.1429	0.2564	21/51	6
-	IRIS2	0.1151	0.2144	26/51	6
-	IRIS3	**0.1487**	*0.2682*	19/51	6
-	IRIS4	0.1097	0.2099	27/51	6
-	IRIS5	0.1452	0.2562	20/51	6
3	IRIS1	0.1682	*0.3155*	16/57	4
-	IRIS2	0.1217	0.2245	31/57	5
-	IRIS3	**0.1890**	0.3059	12/57	4
-	IRIS4	0.1777	0.2962	14/57	4
-	IRIS5	0.1734	0.2780	15/57	4
4	IRIS1	0.2166	0.3655	N/A	N/A
-	IRIS2	0.1710	0.3027	N/A	N/A
-	IRIS3	**0.2378**	0.3716	N/A	N/A
-	IRIS4	0.2246	*0.3745*	N/A	N/A
-	IRIS5	0.1982	0.3404	N/A	N/A

We observe that, in all 4 batches, the best MAP scores are obtained by the voting system, validating the value of using insights from several models at the same time. However, this system performs better in terms of Recall during only one batch, indicating it finds less relevant documents but the one retrieved are better ranked among the "top 10".

It is interesting to note that biomedical knowledge incorporation has a positive effect on the MAP results. Indeed, except during batch 2, "IRIS4" performs better than the other PLMs submitted.

We investigated if reducing the input length by ranking sentences by order of relevance would be beneficial for a PLM. The results tend to confirm this hypothesis, as in the majority of cases MAP scores

Table 3
Results of Systems with Input Reduction

Batch	Model	MAP	Recall
1	IRIS1-R	0.1344	0.2566
-	IRIS-R	0.1480	0.2358
2	IRIS1-R	0.1481	0.2672
-	IRIS-R	0.122	0.2596
3	IRIS1-R	0.1746	0.3084
-	IRIS-R	0.1812	0.2757
4	IRIS1-R	0.2232	0.3732
-	IRIS-R	0.2373	0.3633

increase whether we apply the reduction during training or only during inference. Moreover, applying the reduction during training leads (in most cases) to better results in terms of MAP while applying it only for inference seems to mainly enhance the Recall score.

To better understand our results, we conducted additional analyses. Table 4 presents the scores of 3 representative systems for each type of question. There is a huge drop of performance for "IRIS4" on the factoid questions of Batch 2 which partially explains why this system is globally less effective during this batch. We also observe that, locally speaking, "IRIS3" (the voting system) is not always performing better than other systems but still manage to obtain the highest scores overall. This proves its capability to take advantage and balance the scores of different models.

Table 4
MAP scores per question type

	Yes/No			**List**			**Summary**			**Factoid**		
Batch	IRIS1	IRIS3	IRIS4	IRIS1	IRIS3	IRIS4	IRIS1	IRIS3	IRIS4	IRIS1	IRIS3	IRIS4
1	0.162	0.167	0.155	0.071	0.106	0.123	0.216	0.204	0.2	0.1	0.116	0.114
2	0.215	0.187	0.125	0.149	0.177	0.135	0.091	0.127	0.115	0.097	0.094	0.058
3	0.229	0.232	0.188	0.243	0.332	0.325	0.092	0.093	0.094	0.105	0.104	0.113
4	0.305	0.295	0.262	0.175	0.2	0.22	0.094	0.145	0.129	0.25	0.283	0.268

Table 5 shows that the average query length is increasing along batches as well as the mean number of MeSH terms appearing among them. Our models performances seem to follow this trend as the overall results are also increasing along batches. Except for Batch 2, where "IRIS4" has its worst scores, it seems that detecting more biomedical terms enhances the performance of our tagging model.

Table 5
Queries lengths and MeSH terms detected per query

Batch	Mean number of tokens	Mean number of MeSH terms
1	10.06	1.56
2	10.61	1.92
3	10.72	2.05
4	12.04	2.31

5. Conclusion and Perspectives

The strategy implemented to incorporate biomedical knowledge into BERT cross-encoders is promising as it obtains better results than base systems most of the time. This emphasises the importance of helping such models understand the semantic relations between domain specific terms in a document.

Moreover, we show that taking advantage of several model outputs has a strong and positive effect on the results. Indeed, the voting system provides our best scores which were consistent as they follow the trend of other participating systems being better batch after batch.

Finally, we investigated a way to improve publication processing by bypassing the input length limitation. We trained a simple model to rank important sentences to answer the query. It constrains BERT cross-encoder to focus on relevant parts of each document and to ignore sentences that carry less information. This strategy offers better scores in terms of MAP especially when we used it during both training and inference stages.

Although these results are promising, our models do not perform as well as the strongest systems for this task. Thus, there are several axes of improvement we would like to explore. First of all, we showed modifying the input sequence can have a positive effect. We plan to apply a more complex tagging strategy. One may think about adding new special tokens to tag other biomedical concepts from Unified Medical Language System metathesaurus[9], triplets [19, 20] or other important words in the text [21]. In addition, we only used MeSH as a controlled vocabulary. One way to improve the knowledge we infuse into PLMs would be to take into account its tree structure. An idea would be to apply knowledge graph embedding algorithms and combine those representations with the textual embeddings of PLMs [22, 23, 24]. Moreover, it would be wise to replace the voting system with a method that enables a deep interaction between the models such as multi-layer perceptron or mutli-head attention.

References

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al., Retrieval-augmented generation for knowledge-intensive nlp tasks, Advances in Neural Information Processing Systems 33 (2020) 9459–9474.

[2] A. Nentidis, G. Katsimpras, A. Krithara, S. Lima-López, E. Farré-Maduell, M. Krallinger, N. Loukachevitch, V. Davydova, E. Tutubalina, G. Paliouras, Overview of BioASQ 2024: The twelfth BioASQ challenge on Large-Scale Biomedical Semantic Indexing and Question Answering, in: L. Goeuriot, P. Mulhem, G. Quénot, D. Schwab, L. Soulier, G. Maria Di Nunzio, P. Galuščáková, A. García Seco de Herrera, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), 2024.

[3] A. Nentidis, G. Katsimpras, A. Krithara, G. Paliouras, Overview of BioASQ Tasks 12b and Synergy12 in CLEF2024, in: G. Faggioli, N. Ferro, P. Galuščáková, A. García Seco de Herrera (Eds.), Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, 2024.

[4] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[5] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in: M. Walker, H. Ji, A. Stent (Eds.), Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 2227–2237. URL: https://aclanthology.org/N18-1202. doi:10.18653/v1/N18-1202.

[6] R. Tinn, H. Cheng, Y. Gu, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Fine-tuning large neural language models for biomedical natural language processing, CoRR abs/2112.07869 (2021). URL: https://arxiv.org/abs/2112.07869. arXiv:2112.07869.

[9] https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/index.html

[7] M. Yasunaga, J. Leskovec, P. Liang, Linkbert: Pretraining language models with document links, 2022. arXiv:2203.15827.

[8] K. r. Kanakarajan, B. Kundumani, M. Sankarasubbu, BioELECTRA:pretrained biomedical text encoder using discriminators, in: D. Demner-Fushman, K. B. Cohen, S. Ananiadou, J. Tsujii (Eds.), Proceedings of the 20th Workshop on Biomedical Language Processing, Association for Computational Linguistics, Online, 2021, pp. 143–154. URL: https://aclanthology.org/2021.bionlp-1.16. doi:10.18653/v1/2021.bionlp-1.16.

[9] M. Lesavourey, G. Hubert, Bioasq 11b: Integrating domain specific vocabulary to bert-based model for biomedical document ranking., in: CLEF (Working Notes), 2023, pp. 145–151.

[10] R. F. Nogueira, W. Yang, K. Cho, J. Lin, Multi-stage document ranking with BERT, CoRR abs/1910.14424 (2019). URL: http://arxiv.org/abs/1910.14424. arXiv:1910.14424.

[11] J. Guo, Y. Cai, Y. Fan, F. Sun, R. Zhang, X. Cheng, Semantic models for the first-stage retrieval: A comprehensive review, ACM Trans. Inf. Syst. 40 (2022). URL: https://doi.org/10.1145/3486250. doi:10.1145/3486250.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[13] S. Robertson, H. Zaragoza, et al., The probabilistic relevance framework: Bm25 and beyond, Foundations and Trends® in Information Retrieval 3 (2009) 333–389.

[14] J. Lin, X. Ma, S.-C. Lin, J.-H. Yang, R. Pradeep, R. Nogueira, Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 2356–2362. URL: https://doi.org/10.1145/3404835.3463238. doi:10.1145/3404835.3463238.

[15] 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, CoRR abs/2007.15779 (2020). URL: https://arxiv.org/abs/2007.15779. arXiv:2007.15779.

[16] J. Lin, R. F. Nogueira, A. Yates, Pretrained transformers for text ranking: BERT and beyond, CoRR abs/2010.06467 (2020). URL: https://arxiv.org/abs/2010.06467. arXiv:2010.06467.

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

[18] J. Zhan, J. Mao, Y. Liu, J. Guo, M. Zhang, S. Ma, Optimizing dense retrieval model training with hard negatives, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 1503–1512.

[19] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, Q. Liu, Ernie: Enhanced language representation with informative entities, arXiv preprint arXiv:1905.07129 (2019).

[20] H. Kilicoglu, D. Shin, M. Fiszman, G. Rosemblat, T. C. Rindflesch, Semmeddb: a pubmed-scale repository of biomedical semantic predications, Bioinformatics 28 (2012) 3158–3160.

[21] L. Boualili, J. G. Moreno, M. Boughanem, Highlighting exact matching via marking strategies for ad hoc document ranking with pretrained contextualized language models, Information Retrieval Journal 25 (2022) 414–460.

[22] Q. Dong, Y. Liu, S. Cheng, S. Wang, Z. Cheng, S. Niu, D. Yin, Incorporating explicit knowledge in pre-trained language models for passage re-ranking, in: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2022, pp. 1490–1501.

[23] Q. Xie, P. Tiwari, S. Ananiadou, Knowledge-enhanced graph topic transformer for explainable biomedical text summarization, IEEE Journal of Biomedical and Health Informatics 28 (2024) 1836–1847. doi:10.1109/JBHI.2023.3308064.

[24] J. Tan, J. Hu, S. Dong, Incorporating entity-level knowledge in pretrained language model for biomedical dense retrieval, Computers in Biology and Medicine 166 (2023) 107535.