Hybrid First-stage Retrieval Models for Biomedical Literature

Ji Ma, Ivan Korotkov, Keith Hall, and Ryan McDonald

Google Research
{maji,ivankr,kbhall,ryanmcd}@google.com

Abstract. We describe a hybrid first-stage retrieval model evaluated on BioASQ 8 document retrieval. We show that a hybrid model consistently outperforms comparable neural and term-based models. To train both the hybrid and neural models, we rely on data augmentation, specifically question generation over the Pubmed corpus. In addition to reporting the official runs of this model from BioASQ, we also report some post-challenge improvements. With these improvements, our hybrid model is competitive with the top-scoring systems. When adding a simple neural BERT-based reranker, the model outperforms all systems, on average, across all five batches. This highlights the efficacy of hybrid first-stage retrieval models.

1 Introduction

The BioASQ challenge organizes shared-tasks for semantic understanding of biomedical literature, including document and snippet retrieval, semantic indexing, question answering and summarization [22]. Here, we describe the technical details of an entry to the document retrieval sub-task (Task B Phase A). Specifically, our submissions consist of the following contributions:

Hybrid first-stage retrieval. We use a principled approach to create a sparse-dense retrieval model that combines the benefits of both neural and term-based models. Our term-based model is a standard BM25 model [21] and our neural model falls into the category of dense vector retrieval, which is also known as dual encoder models [1, 19, 5, 2, 7]. We show that since both of these retrieval paradigms can be cast as vector similarity via nearest neighbor search, that a principled hybrid model can be constructed. The neural, term and hybrid models are described in Sections 2–4.

Data Augmentation via Question Generation. Our neural first-stage model requires supervised training data. However, there is a lack of such data for the biomedical domain outside of the few thousand examples from previous BioASQ

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0). CLEF 2020, 22-25 September 2020, Thessaloniki, Greece.

challenges. To address this we use data augmentation [24]. Specifically, we follow the work of Ma et al. [14] and train a question generator on a community QA dataset. We then apply this to Pubmed abstracts to create biomedical-specific pairs of questions and relevant documents. This is described in Section 2.2.

Second-stage reranking. The focus of our contribution was to measure the efficacy of neural first-stage retrieval models for biomedical literature. However, we also experiment with adding a simple BERT-based [4] cross-attention reranker, which has become standard in the IR literature [18, 15, 26], including past BioASQ challenges [20].

2 Neural First-stage Retrieval

Our retrieval model consists of two components. A dense model, which is based on dual encoders [1, 5], aims to capture semantic similarity between query and relevant documents. A sparse model, which is based on term matching, aims at capturing lexical similarity between query and documents. This section focuses on the dense model, and the we describe the sparse model in the next section.

2.1 Dual Encoder

Formally, a dual encoder model consists of two encoders, $\{f_Q(), f_P()\}$ and a similarity function, sim(). An encoder is a function f that takes an item x as input and outputs a real valued vector as the encoding. The similarity function, sim(), takes two encodings, $\mathbf{q}, \mathbf{p} \in \mathbb{R}^N$ and calculates a real valued score, $s = \text{sim}(\mathbf{q}, \mathbf{p})$.

For BioASQ competition, we are interested in encoding natural language texts into real valued vectors. Thus, following recent success in natural language processing, we implement both the two encoders with BERT [4]. In particular, our encoder feeds the input query (or document) string to the BERT model. Then it projects the [CLS] token representation from BERT outputs to a 768-dimensional vector, as the encoding of that query (or document). In addition, we share parameters between query and document encoder, so called Siamese networks [1], which we found consistently improve retrieval performance while reducing the total number of model parameters. We use dot-product as the similarity function. In our initial experiments, we observe no meaningful difference in retrieval performance between dot-product and cosine similarity function.

We train model parameters using softmax cross-entropy loss together with in-batch negatives, i.e., given a query in a batch of (query, relevant-passage) pairs, passages from other pairs are considered irrelevant for that query. In-batch negatives has been widely adopted in training neural network based retrieval models as it enables efficient training via computation sharing [27, 5, 7].

2.2 Question Generation

A major bottleneck in building high accuracy neural retrieval system is the lack of large scale training data. The problem is exacerbated when it comes to specialized domains such as biomedical domain. To handle the data scarcity issue, we follow the approach proposed by Ma et al. [14] which automatically generates synthetic questions on the target domain. Specifically, a transformer-based [23] encoder-decoder generation model is trained to generate questions specific to a given passage. The training data for the generator comprises question-answer pairs mined from community resources such as StackExchange[1] and Yahoo! Answers[2]. When training completes, the question generator is then applied to the target domain document/passage to generate large amount of synthetic queries, in this case Pubmed. Finally, the synthetic question is paired with the passage from which it was generated to form a training example for the dual encoder model.

In this work, our implementation of the question generator follows exactly the same setting as the base model in [14], e.g., both the encoder and decoder consist of 3 transformer layers, parameters between encoder and decoder are shared and are initialized with RoBERTa [12] checkpoint. We refer the reader to the original paper for more details.

2.3 Nearest Neighbour Inference

To serve the dual-encoder retrieval model over a collection of passages, we first run the encoder over every passage offline to create a distributed lookup-table as a backend. At inference, we only need to run the question encoder on the input query. The query encoding is used to perform nearest neighbour search against the passage encodings in the backend. Since the total number of passages is in the order of millions and each passage is projected to a 768 dimensional vector, we use distributed brute-force search for exact inference instead of approximate nearest neighbour search [11, 6].

3 Term-based Retrieval as Nearest Neighbour Search

Term-based retrieval models, such as BM25 [21], have been extensively studied for document retrieval. In fact, for first-stage retrieval, there is significant evidence that term-based models are extremely effective baselines [10]. Term-based models usually use inverted-indexes for inference, taking advantage of lexical sparsity per-document to optimize retrieval speed and memory usage [16]. In this section, we show that inference in term-matching based models, specifically BM25, can be cast as vector dot-product similarity, which will enable principled hybrid models (Section 4).

[1] archive.org/details/stackexchange
[2] webscope.sandbox.yahoo.com/catalog.php?datatype=l

Let Q and P denote a query and a passage, respectively. The BM25 score between Q and P is computed as:

$$\text{BM25}(Q,P) = \sum_{i=1}^{n} \frac{\text{IDF}(q_i) * \text{cnt}(q_i,P) * (k+1)}{\text{cnt}(q_i,P) + k * (1 - b + b * \frac{m}{m_{\text{avg}}})},$$

where q_i are tokens from Q, $\text{cnt}(q_i,\ P)$ is q_i's term frequency in P, k/b are BM25 hyperparameters, IDF is the term's inverse document frequency from the corpus, n/m are the number of tokens in Q/P, and m_{avg} is the collection's average passage length. This can be written as a vector space model. To see this, let $\mathbf{q}^{\text{bm25}} \in [0,1]^{|V|}$ be a $|V|$-dimensional binary encoding of Q, i.e., $\mathbf{q}^{\text{bm25}}[i]$ is 1 if the i-th entry of vocabulary V is in Q, 0 otherwise. Furthermore, let $\mathbf{p}^{\text{bm25}} \in \mathbb{R}^{|V|}$ be a sparse real-valued vector where,

$$\mathbf{p}_i^{\text{bm25}} = \frac{\text{IDF}(p_i) * \text{cnt}(p_i,P) * (k+1)}{\text{cnt}(p_i,P) + k * (1 - b + b * \frac{m}{m_{\text{avg}}})}.$$

We can see that,

$$\text{BM25}(Q,P) = \langle \mathbf{q}^{\text{bm25}}, \mathbf{p}^{\text{bm25}} \rangle$$

Here $\langle , \rangle$ denote vector dot-product.

4 Hybrid First-stage Retrieval

Although dual encoder models are good at capturing semantic similarity, e.g., "Theresa May" and "Prime Minister" [3], we observe lexical matching consistently poses a challenge for first-stage neural retrieval models. For instance, if we consider the question *"Which are the additions of the JASPAR 2016 open-access database of transcription factor binding profiles?"* from a prior year's BioASQ challenge, our initial neural model retrieved this document as the most relevant (title only),

JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles.

whereas a BM25 system returns the much more relevant document,

JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles.

Thus, neural models tend to generalize better than term-models, but term models are advantageous in situations where exact lexical matching is preferable.

In order to build a system that combines the benefits of both neural and term-based retrieval, we combine our neural dual encoder models with BM25 in a principled way. Specifically, leveraging the vector similarity view of BM25 (Section 3) gives rise to a simple hybrid,

$$\begin{aligned}
\text{sim}(\mathbf{q}^{\text{hyb}}, \mathbf{p}^{\text{hyb}}) &= \langle \mathbf{q}^{\text{hyb}}, \mathbf{p}^{\text{hyb}} \rangle \\
&= \langle [\lambda \mathbf{q}^{\text{bm25}}, \mathbf{q}^{\text{nn}}], [\mathbf{p}^{\text{bm25}}, \mathbf{p}^{\text{nn}}] \rangle \\
&= \lambda \langle \mathbf{q}^{\text{bm25}}, \mathbf{p}^{\text{bm25}} \rangle + \langle \mathbf{q}^{\text{nn}}, \mathbf{p}^{\text{nn}} \rangle,
\end{aligned}$$

where $\mathbf{q}^{\text{hyb}}$ and $\mathbf{p}^{\text{hyb}}$ are the hybrid encodings that concatenate the BM25 ($\mathbf{q}^{\text{bm25}}/\mathbf{p}^{\text{bm25}}$) and the neural encodings ($\mathbf{q}^{\text{nn}}/\mathbf{p}^{\text{nn}}$, from Sec 2); and λ is a interpolation hyperparameter that trades-off the relative weight of BM25 versus neural models.

Thus, we can implement BM25 and our hybrid model as nearest neighbor search with hybrid sparse-dense vector dot-product [25]. Note that this results in exact retrieval and not approximate retrieval through post-hoc rescoring, the latter having been studied previously [17, 13, 7]

5 Experiments

Our document collection contains the abstracts of articles from MEDLINE. We discard about 10M abstracts that only contains a title, which leaves us about 18M abstracts. For the dual encoder model, all passages are truncated at 300 wordpiece tokens with BERT tokenization.

All evaluation is done either by the BioASQ challenge via uploaded results, or subsequently using the official BioASQ evaluation script. As per challenge rules, we returned at most 10 relevant documents per question.

5.1 Systems

BM25 We build a standard BM25 retrieval system based on IDF values computed on the document collection. This is a unigram model using the bioclean tokenization script from BioASQ.

DE This the dual encoder model described in section 2.1, which is based on a pretrained BERT model. In this work, we create our own wordpiece vocabulary on pubmed abstracts with 107137 entries. Our BERT model consists of 12 transformer [23] layers, each with hidden size 1024 and 16 attention heads. We use the same sentence sampling procedure as reported in the original BERT paper, e.g., the combined sequence has length no longer than 512 tokens, and we uniformly mask 15% of the tokens from each sequence for masked language model prediction. We update the next sentence prediction task by replacing original binary-cross-entropy loss with softmax cross-entropy loss as described in 2.1. We use the same hyper-parameter values for BERT pretraining except that the learning rate is set 2e-5, and the model is trained for 300,000 steps.

To train the dual encoder model, we use supervised data provided by BioASQ, as well as synthetic data generated using method mentioned in section 2.2. For supervised data, we use BioASQ 8B training data where the last 200 questions are used as development set. The synthetic data contains about 103,635,592 question-passage pairs where questions are generated from pubmed abstracts. The dual encoder model is trained with a batch size of 6144. For each batch, 20% of the examples come from synthetic data, and the rest come from supervised data. We train the model for 100,000 steps using Adam [8] with a learning rate 5e-6, $\beta 1 = 0.9$, $\beta 2 = 0.999$. Similar to BERT pretraining, we also apply L2 weight decay of 0.01, and warm up learning rate for the first 10,000 steps.

Hybrid This is identical to DE, but instead of using the pure neural model, we train the hybrid model in section 4 with $\lambda = 1.5$ which is achieved by running a grid search on the development set.

HybridRerank This system applies a reranker on top of the output from the Hybird system. We cast the reranking to logistic regression problem: given a question-passage pair, the model predicts whether that passage is relevant to the question or not. Here a passage is the concatenation of an article title with the abstract of that article. The reranking model is also based on BERT, i.e., we concatenate the query and passage as the input for BERT and apply a MLP on top of the [CLS] token representation. We use question-passage pairs from BioASQ 8B as positive examples. Negative examples are created using the same queries but with passages returned by the BM25 system. We train the model for 1 epoch, with the same hyper parameter values as used to train the dual encoder model. For inference, given a query, we sort the top 10 output from the Hybrid system in descending order according to their reranking score.

5.2 Official Results

Official results for our submissions are shown in Table 1. Not all systems were submitted to all batches. We report only Mean Average Precision (MAP) as it is the official metric for the document retrieval challenge. These are the preliminary results before human judgements, which are still outstanding. A number of things can be observed:

1. BM25 is signiciantly better than our neural DE model. As mentioned previously, BM25 is a very strong baseline. However, we suspect that part of this is due to the nature of the BioASQ data, where relevance annotators are also who create the questions. This has been shown to bias datasets in favor of term-based results [9]. This is exacerbated for the preliminary results, where relevance judgements are gathered via pre-existing search tools like Pubmed, which themselves are heavily biased to term matching.
2. The Hybrid model consistently outperforms the BM25 model – by about 2pts on average. This shows that hybrid retrieval is a very viable approach to first-stage retrieval for biomedical literature.
3. Adding a BERT-based cross-attention reranker consistently increases accuracy, by 1-3pts. This is consistent with previous studies in the domain [20].
4. Our final reranking model is competitive with the best scoring systems on the batches in which it was scored. Given the simplicity of the model, we expect that further optimizations will increase accuracy further. E.g., the best scoring system from last year – and one of the top systems from this year – used a joint document-snippet model [20].

	Batch 2	Batch 3	Batch 4	Batch 5
	MAP	MAP	MAP	MAP
BM25	0.2718	0.3877	0.3631	0.4287
DE	0.1173	0.2756	0.2600	0.3190
Hybrid	0.2806	0.3995	0.3866	0.4437
HybridRerank	– –	0.4303	0.4121	0.4593
Best Reporting System	0.3304	0.4510	0.4163	0.4842

Table 1. Mean average precision (MAP) official results for batches 2–5.

	Batch 1	Batch 2	Batch 3	Batch 4	Batch 5	Average
	MAP	MAP	MAP	MAP	MAP	MAP
BM25	0.3538	0.2955	0.3891	0.3872	0.4286	0.3708
DE	0.2661	0.1869	0.3096	0.2771	0.3190	0.2717
Hybrid	0.3711	0.3087	0.4141	0.4099	0.4437	0.3895
HybridRerank	0.3877	0.3226	0.4235	0.4351	0.4592	0.4056
Best Reporting System	0.3398	0.3304	0.4510	0.4163	0.4842	0.4043

Table 2. Mean average precision (MAP) updated results for batches 1–5.

5.3 Updated Results

While the BioASQ challenge was underway, we updated our models and data to improve them. Here we report results for updated models that incorporate these improvements. These are not official submissions, but use the BioASQ evaluation script and are thus comparable to official results.

We made the following updates:

1. *Data fix.* After batch 4, we realized that our data pipeline sometimes did not include the full abstract. This was fixed.
2. *Bigram BM25.* Our original BM25 model was a unigram model that used the bioclean tokenizer supplied by BioASQ. We tried using a BERT-based tokenizer (the same one as used by the DE model) and found that this performed better.
3. *Better abstract modeling for DE.* For our DE model, we originally truncated abstracts at 300 wordpieces. Instead we divide the abstract into blocks, each 300 wordpieces in length and index each seperately. At inference, if two blocks from the same abstract are returned, we remove the duplicate document.

These updates were incorporated and all the new models were run on all batches. Table 2 shows the results relative to the best system per batch, as well as the average across all five batches. Compared to Table 1, we can see that all numbers go up and now the HybridRerank system is the top system on two batches and overall on average. We should note that the 'Best Reporting System' row is not the same submission across batches, as different systems (and teams) performed best depending on the batch. Thus, the average of this row does not represent a single system, but the average over possibly many systems.

6 Conclusions

In this paper we described our submissions to the BioASQ challenge. Specifically, we show that hybrid term-neural models are a viable first-stage retrieval method. For the neural portion, using data augmentation techniques as proposed by Ma et al. [14] were required to attain reasonable performance and are likely necessary in cases where there is little supervised data.

Overall, our methods were competitive, especially when combined with a re-ranker. Post challenge improvements around data quality and minor modeling changes (e.g., bigram BM25) pushed the results near the top of the challenge, highlighting the effectiveness of our models.

References

1. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a "siamese" time delay neural network. In: Advances in neural information processing systems. pp. 737–744 (1994)
2. Chang, W.C., Yu, F.X., Chang, Y.W., Yang, Y., Kumar, S.: Pre-training tasks for embedding-based large-scale retrieval. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=rkg-mA4FDr
3. Cohen, D., Mitra, B., Hofmann, K., Croft, W.B.: Cross domain regularization for neural ranking models using adversarial learning. CoRR **abs/1805.03403** (2018)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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). pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019). https://doi.org/10.18653/v1/N19-1423, https://www.aclweb.org/anthology/N19-1423
5. Gillick, D., Presta, A., Tomar, G.S.: End-to-end retrieval in continuous space. CoRR **abs/1811.08008** (2018)
6. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with gpus. arXiv preprint arXiv:1702.08734 (2017)
7. Karpukhin, V., Oğuz, B., Min, S., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. CoRR (04 2020)
8. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. International Conference on Learning Representations (12 2014)
9. Lee, K., Chang, M.W., Toutanova, K.: Latent retrieval for weakly supervised open domain question answering. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 6086–6096. Association for Computational Linguistics, Florence, Italy (Jul 2019). https://doi.org/10.18653/v1/P19-1612, https://www.aclweb.org/anthology/P19-1612
10. Lin, J.: The neural hype and comparisons against weak baselines. In: ACM SIGIR Forum. ACM New York, NY, USA (2019)
11. Liu, W., Wang, J., Kumar, S., Chang, S.F.: Hashing with graphs. In: Proceedings of the International Conference on Machine Learning (2011)
12. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized BERT pretraining approach. CoRR **abs/1907.11692** (2019)

13. Luan, Y., Eisenstein, J., Toutanova, K., Collins, M.: Sparse, dense, and attentional representations for text retrieval. arXiv preprint arXiv:2005.00181 (2020)
14. Ma, J., Korotkov, I., Yang, Y., Hall, K., McDonald, R.: Zero-shot neural retrieval via domain-targeted synthetic query generation (2020)
15. MacAvaney, S., Yates, A., Cohan, A., Goharian, N.: Cedr: Contextualized embeddings for document ranking. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (2019)
16. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to information retrieval. Cambridge university press (2008)
17. McDonald, R., Brokos, G., Androutsopoulos, I.: Deep relevance ranking using enhanced document-query interactions. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 1849–1860. Association for Computational Linguistics, Brussels, Belgium (Oct-Nov 2018). https://doi.org/10.18653/v1/D18-1211, https://www.aclweb.org/anthology/D18-1211
18. Nogueira, R., Cho, K.: Passage re-ranking with bert. arXiv preprint arXiv:1901.04085 (2019)
19. Palangi, H., Deng, L., Shen, Y., Gao, J., He, X., Chen, J., Song, X., Ward, R.: Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval. IEEE/ACM Transactions on Audio, Speech, and Language Processing **24**(4), 694–707 (2016)
20. Pappas, D., McDonald, R., Brokos, G.I., Androutsopoulos, I.: AUEB at BioASQ 7: document and snippet retrieval. In: Proceedings of the BioASQ Workshop (2019)
21. Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M.M., Gatford, M.: Okapi at trec-3. In: Overview of the Third Text REtrieval Conference (TREC-3). pp. 109–126. Gaithersburg, MD: NIST (January 1995), https://www.microsoft.com/en-us/research/publication/okapi-at-trec-3/
22. Tsatsaronis, G., Balikas, G., Malakasiotis, P., Partalas, I., Zschunke, M., Alvers, M.R., Weissenborn, D., Krithara, A., Petridis, S., Polychronopoulos, D., Almirantis, Y., Pavlopoulos, J., Baskiotis, N., Gallinari, P., Artiéres, T., Ngomo, A.C.N., Heino, N., Gaussier, E., Barrio-Alvers, L., Schroeder, M., Androutsopoulos, I., Paliouras, G.: An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. BMC bioinformatics **16**, 138 (April 2015). https://doi.org/10.1186/s12859-015-0564-6
23. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems, pp. 5998–6008. Curran Associates, Inc. (2017), http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf
24. Wong, S.C., Gatt, A., Stamatescu, V., McDonnell, M.D.: Understanding data augmentation for classification: When to warp? In: 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA). pp. 1–6 (2016)
25. Wu, X., Guo, R., Simcha, D., Dopson, D., Kumar, S.: Efficient inner product approximation in hybrid spaces. arXiv preprint arXiv:1903.08690 (2019)
26. Yang, W., Zhang, H., Lin, J.: Simple applications of bert for ad hoc document retrieval. arXiv preprint arXiv:1903.10972 (2019)
27. Yih, W.t., Toutanova, K., Platt, J.C., Meek, C.: Learning discriminative projections for text similarity measures. In: Proceedings of the Fifteenth Conference on Computational Natural Language Learning. pp. 247–256. Association for Computational Linguistics, Portland, Oregon, USA (Jun 2011), https://www.aclweb.org/anthology/W11-0329