PASH at TREC 2021 Deep Learning Track: Generative Enhanced Model for Multi-stage Ranking

Yixuan Qiao[1,*], Hao Chen[1], Tuozhen Liu[2], Xianbin Ye[3], Jun Wang[1], Peng Gao[1], and Guotong Xie[1,4,5,*]

[1] Ping An Health Technology, Beijing, China
[2] Central University of Finance and Economics, Beijing, China
[3] Jinan University, Guangzhou, China
[4] Ping An Health Cloud Company Limited., Shenzhen, China
[5] Ping An International Smart City Technology Co., Ltd., Shenzhen, China

Abstract. This paper describes the PASH participation in TREC 2021 Deep Learning Track. In the recall stage, we adopt a scheme combining sparse and dense retrieval method. In the multi-stage ranking phase, point-wise and pair-wise ranking strategies are used one after another based on model continual pre-trained on general knowledge and document-level data. Compared to TREC 2020 Deep Learning Track, we have additionally introduced the generative model T5 to further enhance the performance.

Keywords: expansion method · dense retrieval · transfer learning · multi-stage ranking · model parallel.

1 Methodology

In this section, we briefly describe the components in the multi-stage ranking pipeline. Details of some methods can be found in TREC 2020 notebook paper. The passage ranking task contains 285,328 queries on a collection of 138,364,198 passages, totally 296,419 query passage pairs annotated as positive for relevance. Few queries are matched with multiple relevant passages. The document ranking task contains 331,748 queries on a collection of 11,959,635 documents, totally 341,836 query document pairs annotated as positive for relevance. Few queries are matched with multiple relevant documents. NDCG is the main official evaluation index.

1.1 Multi-way Matching

Sparse retrieval We use the docTTTTTquery[1] (also known as docT5query or doc2query-T5) to generate queries for which the passage might be relevant.

*Corresponding Author. Email: {qiaoyixuan528, xieguotong}@pingan.com.cn.

For passage, We sample 40 queries per passage with T5-base. For document, we first segment each document into passages by applying a sliding window of ten sentences with a stride of five. Each passage is then prepended with the url and title of the document. Finally, we generate ten queries per passage. Each document prepended with the url and title info was appended with all its predicted queries are indexed by BM25[2] as before.

Dense retrieval Dense retrieval methods have shown great promise over sparse retreval methods in a range of NLP problems. We choose ColBERT[3] as our Dense Retriever due to the effectiveness/efficiency tradeoffs.

1.2 Multi-stage Ranking

We inherit the model from TREC 2020 Deep Learning Track including BERT-Large[4], ALBERT-XXLarge[5], ELECTRA-Large[6] and XLNet-Large[7] model. In particular, BERT-Large uses the same pre-training and fine-tuning strategies as last year but using this year's data set. Compared with last year, we have made the following further improvements:

- We use pair-wise loss as the goal of the second stage of ranking. So we can use more data instead of TREC labeled 4-class data.
- Generative model T5[8] is introduced to further enhance the performance
  a) Implementation based on Megatron-LM[9] repository.
  b) Same training strategy as [10].
  c) Training the T5-3B model takes approximately 12 days on 32 V100 GPUs consuming about 12.8M training examples.
  d) The T5-11B model takes about 30 days under the same situation. An amazing fact is that the zero-shot NDCG@5 of the T5-11B is already 0.3651.

1.3 Ensemble

Benefited from multi-way matching, we just sequentially train different re-rankers with different random seeds for ensemble learning.

2 Results

We submitted six official runs in each passage and document ranking subtask. Table 1 and 2 present the results of our runs in passage ranking task. Table 3 and 4 present the results of our runs in document ranking task. We use _r∗ to denote the runs for re-ranking, _f∗ for full ranking.

Table 1. Results on the TREC 2021 passage re-rank task.

RUN	MAP@100	NDCG@5	NDCG@10	R@100
pash_r1	0.2362	0.7190	0.6951	0.3261
pash_r2	0.2389	0.7390	0.7076	0.3261
pash_r3	0.2385	0.7390	0.7072	0.3261

Table 2. Results on the TREC 2021 passage full rank task.

RUN	MAP@100	NDCG@5	NDCG@10	R@100
pash_f1	0.3193	0.7596	0.7494	0.4868
pash_f2	0.3318	0.7596	0.7494	0.5300
pash_f3	0.3378	0.7596	0.7494	0.5500

Table 3. Results on the TREC 2021 document re-rank task.

RUN	MAP@100	NDCG@5	NDCG@10	R@100
pash_doc_r1	0.2665	0.7452	0.7150	0.3195
pash_doc_r2	0.2640	0.7308	0.7076	0.3195
pash_doc_r3	0.2672	0.7383	0.7164	0.3195

Table 4. Results on the TREC 2021 document full rank task.

RUN	MAP@100	NDCG@5	NDCG@10	R@100
pash_doc_f1	0.3111	0.6745	0.5832	0.3683
pash_doc_f4	0.3498	0.7516	0.7404	0.4412
pash_doc_f5	0.3521	0.7508	0.7368	0.4419

3 Conclusion

The experiments demonstrate that the expressiveness of ColBERT is well integrated with BM25+docTTTTTquery. Compared with the transformer-based model, the generative model T5 has a higher NDCG@5 and a lower NDCG@10, which means that relevant passages are already in top5. This phenomenon largely hints at the thirst for multi-stage ranking in generative models, which can be expected to show powerful effects with the pair-wise or list-wise training. We believe that the interweaving sequence of different methods will be a focus point of future work.

References

1. Nogueira, R., Yang, W., Lin, J. & Cho, K. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375* (2019).
2. Zeng, Z. & Sakai, T. Bm25 pseudo relevance feedback using anserini at waseda university. In *OSIRRC@ SIGIR*, 62–63 (2019).
3. Khattab, O. & Zaharia, M. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 39–48 (2020).
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)*, 4171–4186 (2019).
5. Lan, Z. *et al.* Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations* (2019).
6. Clark, K., Luong, M.-T., Le, Q. V. & Manning, C. D. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555* (2020).
7. Yang, Z. *et al.* Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems* **32** (2019).
8. Raffel, C. *et al.* Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* **21**, 1–67 (2020).
9. Narayanan, D. *et al.* Efficient large-scale language model training on gpu clusters using megatron-lm. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 1–15 (2021).
10. Pradeep, R., Nogueira, R. & Lin, J. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *arXiv preprint arXiv:2101.05667* (2021).