Joint Inference of Retrieval and Generation for Passage Re-ranking

Wei Fang and **Yung-Sung Chuang** and **James Glass**
Massachusetts Institute of Technology
{weifang,yungsung,glass}@mit.edu

Abstract

Passage retrieval is a crucial component of modern open-domain question answering (QA) systems, providing information for downstream QA components to generate accurate and transparent answers. In this study we focus on passage re-ranking, proposing a simple yet effective method, *Joint Passage Re-ranking* (JPR), that optimizes the mutual information between query and passage distributions, integrating both cross-encoders and generative models in the re-ranking process. Experimental results demonstrate that JPR outperforms conventional re-rankers and language model scorers in both open-domain QA retrieval settings and diverse retrieval benchmarks under zero-shot settings.[1]

1 Introduction

Passage retrieval is a crucial component in open-domain question answering (QA) (Chen and Yih, 2020), a task that requires answering questions from a wide range of domains and could be applied in systems that fulfill user's information needs (Voorhees et al., 1999). Retrieval offers downstream QA systems grounding information, which not only improves accuracy in a lot of cases but also provides transparency to how systems generate answers, similar to how articles provide references and citations, such that model hallucinations can be checked with ease. Furthermore, the set of documents to be retrieved from, or knowledge base, can be quickly updated with new documents and knowledge such that models can adapt to temporal changes, and do not need to be continuously re-trained nor require online training paradigms for continual learning.

Early retrieval methods are typically based on term-matching, such as BM25 (Robertson et al., 2009) or TF-IDF (Salton et al., 1975). Such methods, called sparse retrievers, perform keyword matching efficiently with an inverted index to find relevant contexts. Sparse retrievers often achieve reasonable performance while being computationally efficient and does not require training, but are shown to have limited abilities beyond lexical matching.

Recently, dense retrievers that encode text with continuous embeddings have been heavily studied and utilized in contemporary QA systems, often outperforming their sparse counterparts on high resource evaluation settings (Karpukhin et al., 2020). There are a few drawbacks however, such as higher computational demands during both training and inference, inability to handle large contexts (Luan et al., 2021), and difficulty in generalizing to new domains especially those with limited data (Reddy et al., 2021). Hybrid methods have been explored to get the best of both worlds, generally utilizing an efficient sparse method to retrieve a larger number of possibly relevant contexts, and then perform passage re-ranking with a more computationally-intensive dense model for refined scoring (Nogueira and Cho, 2019).

In this work, we focus on passage re-ranking and explore the use of generative models alongside conventional re-rankers. Previous work have explored pre-trained language models (LM) as the re-ranking scorer (Sachan et al., 2022), however we find that it underperforms conventional re-rankers for both supervised and zero-shot settings. Starting from maximizing mutual information (MI) for inference, which measures how much more queries and passages co-occur compared to appearing independently, we show how a small generative model can be effectively used with conventional cross-encoding re-rankers for improved performance. Experiments on a supervised setting for open-domain QA retrieval and a zero-shot setting across a suite of diverse retrieval benchmarks validate our approach. Our contributions can be summarized as follows:

[1] Source code is available at https://github.com/wfangtw/jpr

Findings of the Association for Computational Linguistics: EACL 2024, pages 2289–2298

- We propose *Joint Passage Re-ranking* (JPR), a method utilizing both a cross-encoder and a generative model in the retrieval re-ranking process, optimizing the mutual information between query and passage distributions.
- We demonstrate that JPR outperforms conventional re-rankers and generative scorers in open-domain QA retrieval evaluation and diverse zero-shot retrieval datasets.

2 Joint Passage Re-ranking (JPR)

Consider the two distributions $p(\boldsymbol{x})$ and $p(\boldsymbol{z})$ over all queries $\boldsymbol{x} \in \mathcal{X}$ and all passages $\boldsymbol{z} \in \mathcal{Z}$. The conditional distributions $p(\boldsymbol{z}|\boldsymbol{x})$ and $p(\boldsymbol{x}|\boldsymbol{z})$ can be used to infer one domain based on the other. The joint distribution $p(\boldsymbol{x}, \boldsymbol{z})$ characterizes the combined structure of both domains, where $p(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{x})p(\boldsymbol{z}|\boldsymbol{x}) = p(\boldsymbol{z})p(\boldsymbol{x}|\boldsymbol{z})$.

Here $p_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ defines a passage retrieval model, which we parametrize by $\boldsymbol{\phi}$, generally trained with maximum likelihood estimation (MLE): $\mathcal{L}_{\text{retrieval}}(\phi) \triangleq -\mathbb{E}_{x,z\sim p(\boldsymbol{x},\boldsymbol{z})}\left[\log p_{\boldsymbol{\phi}}(z|x)\right]$. During inference, finding the most probable relevant passage can be written as:

$$\hat{z} = \arg\max_{z} \log p_{\boldsymbol{\phi}}(z|x). \tag{1}$$

Since we focus on passage re-ranking, we treat $p_{\boldsymbol{\phi}}(z|x)$ in Eq. 1 as re-ranking scores.

2.1 Inference by Maximizing Mutual Information

In passage retrieval, documents are commonly chunked into multiple passages of fixed length, some of which containing summaries or general information that are often estimated to have high probabilities by retrieval rankers but do not contain specifics regarding the given query. One of such example is shown in Figure 1. In this work, we approach inference by finding the passage that maximizes the *pointwise mutual information* (PMI) between both domains instead of likelihood:

$$\hat{z} = \arg\max_{z} \Big(\log p(z|x) - \log p(z)\Big). \tag{2}$$

We see that maximizing PMI adds a penalizing term compared to MLE in Eq. 1, which discounts such passages that unconditionally have a higher probability, and biases the model towards those that are specific to the given query. A hyperparameter λ is added to control the regularization term. Using

Passage (z)

I Can Only Imagine (film) I Can Only Imagine is a 2018 American Christian drama film directed by the Erwin Brothers and written by Alex Cramer, Jon Erwin, and Brent McCorkle, based on the story behind the MercyMe song of the same name, the best-selling Christian single of all time. The film stars J. Michael Finley as Bart Millard, the lead singer who wrote the song about his relationship with his father (Dennis Quaid). Madeline Carroll, Priscilla Shirer, Cloris Leachman, Trace Adkins and Brody Rose also star. "I Can Only Imagine" was released in the United States on March 16,

Query (x)	$\log p(z\|x)$	label
who produced the movie i can only imagine	-0.882	0
who played amy grant i i can only imagine	-0.913	0
who wrote the country song i can only imagine	-2.466	1
who wrote and performed i can only imagine	-2.682	1
when was i can only imagine the song released	-3.893	0
when is i can only imagine coming out	-4.507	0

Figure 1: Example showing a passage that is estimated to have high retrieval probabilities for multiple queries by a conventional re-ranker. Each query asks about different specifics of a movie, however the passage contains mostly general information, and could not be used to answer several top-ranked questions. This motivates our use of a penalization term to discount these high probability passages that are not specific to the input query.

Bayes' theorem, we can rewrite Eq. 2 as:

$$\begin{aligned}\hat{z} &= \arg\max_{z} \Big(\log p(z|x) - \lambda \log p(z)\Big) \\ &= \arg\max_{z} \Big((1-\lambda)\log p(z|x) + \lambda \log p(x|z)\Big).\end{aligned} \tag{3}$$

The PMI objective is equivalent to the convex combination of the terms $\log p(z|x)$ and $\log p(x|z)$. Notice that the latter term can be viewed as a conditional generation model that gives the probability of generating a query given a passage. We denote the generative model by $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ with parameters $\boldsymbol{\theta}$. This term was previously explored as the sole inference objective in Sachan et al. (2022), in which an LM was used as a question generator for re-scoring. Instead of using either the retrieval model or the generative model only, as explored in prior work, Eq. 3 provides a simple way to use both models jointly for inference, which we refer to as *Joint Passage Re-ranking* (JPR).

2.2 Joint Fine-tuning

A straighforward way to obtain the two models that can be used for the aforementioned MI-based inference is to train both models using MLE seperately. The retrieval model can be trained with $\mathcal{L}_{\text{retrieval}}(\phi)$, while the generative model can be a trained with a

Re-ranking Method	Cross-Encoder? $\log p_{\phi}(z\|x)$	Generative? $\log p_{\theta}(x\|z)$	Natural Questions Top-1	Top-5	Top-10	TriviaQA Top-1	Top-5	Top-10
BM25	✗	✗	22.1	43.8	54.5	46.3	66.3	71.7
BERT-FT	✓	✗	49.4	66.4	71.4	66.7	77.6	80.2
T5-FT	✗	✓	34.3	59.6	66.7	56.8	74.1	78.0
UPR (T0-3B)	✗	✓	36.8	61.6	68.2	57.7	75.4	78.5
JPR	✓	✓	51.0	<u>**68.0**</u>	<u>**72.3**</u>	68.3	78.3	**80.5**
JPR-FT	✓	✓	<u>**51.4**</u>	67.5	71.9	**69.2**	**78.5**	**80.5**
UPR (LLaMA-33B)	✗	✓	35.0	61.5	69.0	57.2	76.7	79.5
JPR (LLaMA-33B)	✓	✓	48.2	66.9	71.5	<u>70.1</u>	<u>79.3</u>	<u>80.8</u>

Table 1: Top-K retrieval accuracy (%) on the Natural Questions and TriviaQA test sets. All non-BM25 methods re-rank the top-100 passages retrieved by BM25. Best overall are in **bold** while best non-LLM are <u>underlined</u>.

simple LM loss $\mathcal{L}_{\text{generation}}(\boldsymbol{\theta})$.

However, the terms in Eq. 3 are derived when the distributions are matched, that is, when $p(\boldsymbol{x})p_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) = p(\boldsymbol{z})p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$. When the two models are optimized independently, we cannot ensure that this holds. We therefore attempt to enforce this constraint with joint fine-tuning. Similar to previous work on dual supervised learning, we approach this by adding a regularization term, defined as the symmetric KL divergence between the two distributions: $\mathcal{L}_{\text{match}}(\boldsymbol{\phi},\boldsymbol{\theta}) \triangleq D_{\text{sym-KL}}\big(p_{\boldsymbol{\phi}}(\boldsymbol{x},\boldsymbol{z})||p_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{z})\big)$, by enforcing alignment of the marginals multiplied by the conditional probabilities. The joint fine-tuning objective is obtained by combining all three losses: $\mathcal{L}(\boldsymbol{\phi},\boldsymbol{\theta}) \triangleq \mathcal{L}_{\text{retrieval}}+\mathcal{L}_{\text{generation}}+\alpha\mathcal{L}_{\text{match}}$, where α is a regularization hyperparameter. The additional fine-tuning aligns the two conditional distributions such that the conditions for our derivations hold, thereby enhancing the overall performance.

3 Experiments

3.1 Open-Domain QA Retrieval

3.1.1 Data

First, we evaluate on two standard open-domain QA retrieval benchmark datasets: Natural Questions (NQ; Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017). Wikipedia passages used in DPR (Karpukhin et al., 2020) were used in these experiments, which consists of 21M 100-word passages from the English Wikipedia dump of Dec. 20, 2018 (Lee et al., 2019). Additional dataset information can be found in Appx. A.

3.1.2 Setup and Baselines

We adopt the setting from prior work using standard dataset splits, retrieving the top 100 passages for re-ranking. We use Pyserini (Lin et al., 2021) for BM25 as the initial retriever, with default Lucene parameters of $k = 0.9$ and $b = 0.4$. We report top-K retrieval accuracy, the standard metric.

We compare JPR against several baselines: 1) cross-encoding re-ranker (BERT-FT), a fine-tuned BERT-based (Devlin et al., 2019) re-ranker, running inference with Eq. 1; 2) generative re-ranker (T5-FT), a fine-tuned T5 conditional generation model (Raffel et al., 2020) with the second term of Eq. 3 as inference objective; and 3) UPR (Sachan et al., 2022), a generative re-ranker using the larger pre-trained T0-3B model (Sanh et al., 2022).

For our approach, we report one setting with joint inference (JPR), and another with joint fine-tuning followed by the MI-based inference (JPR-FT). Joint inference uses the separately fine-tuned retrieval re-ranker and generative re-ranker described above directly. For joint fine-tuning, we bootstrap with the two models, and further fine-tune with our proposed objective to match the discriminative and generative distributions. λ and α are chosen by performance on the development set. Additional details can be found in Appx. B.

Furthermore, we aim to explore the effects of scaling generative re-rankers up. We experiment with a large language model (LLM), the 33B-parameter LLaMA (Touvron et al., 2023), as our generative re-ranker for both UPR and JPR.

3.1.3 Results and Discussion

Open-domain QA retrieval results are shown in Table 1. Using the conventional cross-encoder BERT-FT on initial BM25 results yields decent improvements. UPR, not fine-tuned but being much larger, significantly underperforms BERT-FT. The fine-tuned generative model T5-FT, 15× smaller than the T0-3B model in UPR, nearly matches the

Dataset	BM25	Re-ranking Method					
		BERT-FT	T5-FT	UPR	JPR	UPR (LLM)	JPR (LLM)
TREC-DL 2019	50.8	<u>74.9</u>	<u>65.6</u>	-	<u>75.0</u>	-	-
TREC-COVID	65.6	75.7	75.7	76.5	78.2	76.5	77.2
NFCorpus	32.6	35.0	33.2	34.8	35.3	33.5	35.7
NQ	32.9	53.3	43.8	44.5	52.1	45.3	54.0
HotpotQA	60.3	70.7	68.5	70.9	72.4	72.3	72.1
FiQA-2018	23.6	34.7	35.7	42.0	38.5	40.3	36.6
ArguAna	*41.4*	*41.8*	50.2	*50.9*	49.3	28.5	43.3
Touché-2020	36.7	27.1	25.0	21.0	26.8	18.5	25.7
CQADupStack	29.9	37.1	37.7	40.2	39.7	42.9	39.0
Quora	78.9	82.5	81.2	83.6	84.8	84.4	84.1
DBPedia	31.3	40.9	34.6	35.5	40.5	35.1	41.6
SCIDOCS	15.8	16.6	16.9	17.6	18.3	18.1	17.1
FEVER	75.3	81.8	75.7	61.3	82.5	62.5	79.7
Climate-FEVER	21.3	25.3	18.4	14.6	25.2	11.2	24.9
SciFact	66.5	68.8	69.3	70.4	72.7	65.7	70.3
Average	43.7	49.4	47.6	47.4	**51.2**	45.3	50.1

Table 2: Zero-shot results on BEIR, scores denote **nDCG@10**. All methods re-rank the top-100 passages retrieved by BM25, except for TREC-DL 2019 to compare to prior work. Best overall are in **bold**. <u>Underlined</u> indicate in-domain performance, and *italicized* are based on Pyserini reproductions, differing from those reported in prior work.

performance of UPR. When using JPR, which corresponds to scoring with Eq. 3 using the re-ranker BERT-FT and the generative model T5-FT, surpasses all baselines. The generative model, although used by itself underperforms BERT-FT, boosts performance especially for the top retrieved passages. Matching distributions (JPR-FT) by fine-tuning for a small amount of steps further improves performance, albeit more modestly. For LLM generative re-ranking, despite being multitudes larger, LLaMA-33B surprisingly underperforms against T5-FT and T0-3B on NQ for both UPR and JPR, however on TriviaQA JPR with LLaMA-33B achieves best overall results. Appx. C shows further results for different model pairings.

3.2 Zero-Shot Retrieval

3.2.1 Data

We further evaluate in a transfer learning setting on BEIR (Thakur et al., 2021), a commonly used benchmark consisting of a suite of information retrieval datasets that span multiple tasks and domains. Datasets in the benchmark contain queries and passages of a variety of styles and lengths, and no training data is provided, making it considerably difficult for models to perform well across all datasets. See Appx. D for more details.

3.2.2 Setup and Baselines

We follow BEIR's zero-shot evaluation on all tasks, using MS MARCO (Nguyen et al., 2017) as training data. Pyserini is used for BM25 to retrieve 100 passages, with default parameters and indexing title and passage as separate fields[2][3]. The Normalized Cumulative Discount Gain (nDCG@K) (Wang et al., 2013) is used for evaluation, with $K = 10$, computed by the official TREC evaluation tool (Van Gysel and de Rijke, 2018).

We compare against the three baselines used previously with slight differences: 1) conventional discriminative re-ranker (BERT-FT), using a BERT-based re-ranker pre-trained on MS MARCO with the same configuration (Reimers and Gurevych, 2019); 2) generative re-ranker (T5-FT), using the same `t5-base-lm-adapt` but fine-tuned on MS MARCO; and 3) UPR, but re-ranked over 100 instead of 1000. For our proposed approach, we only evaluate the joint inference method (JPR), as the MS MARCO pre-trained re-ranker from SBERT[4] is already at a saddle point, and using it to bootstrap leads to degraded performance. Detailed training hyperparameters can be found in Appx. E.

3.2.3 Results and Discussion

Zero-shot results on BEIR are presented in Table 2. JPR attains roughly 2% absolute gain on average simply by utilizing both discriminative and generative models for inference, which is more prominent when compared against in-domain performances in Sec. 3.1 and on TREC-DL 2019. JPR surpasses BERT-FT on 10 out of the 14 tasks and is roughly equal on the other 4, and eclipses T5-FT on 13 of 14. Notably, for two tasks, FEVER and Climate-FEVER, generative re-rankers struggle and exhibit degraded performance, whereas JPR avoids this issue and outperforms BERT-FT. When using the comparatively huge LLaMA, we see that UPR worsens on average, mostly due to major underperformance on tasks such as ArguAna, Touché-2020, FEVER, and Climate-FEVER. On most other tasks it outperforms UPR, suggesting that larger models' effects may scale both ways, positively on familiar tasks, such as CQADupStack which LLaMA had exposure during LM training, and negatively on a few out-of-domain ones. JPR (LLM) can mitigate the worst cases, however it mostly does not

[2]Pyserini reproductions for BEIR can be found at https://castorini.github.io/pyserini/2cr/beir.html.

[3]We follow BEIR and retrieve 100, which is more practical.

[4]https://www.sbert.net/docs/pretrained_cross-encoders.html

outperform JPR that uses the considerably smaller generative model.

4 Related Work

Passage re-ranking seeks to combine the advantages of sparse retrieval methods, such as efficiency, precise matching, and low-resource generalizability (Sciavolino et al., 2021; Reddy et al., 2021), with the superior performance of dense methods in the presence of extensive annotated data (Karpukhin et al., 2020; Guu et al., 2020). Early work by Nogueira and Cho (2019) examined BERT-based supervised re-rankers, while later research proposed reader prediction based re-ranking (Mao et al., 2021) and attempted to use LMs as re-rankers (Sachan et al., 2022), although with limitations. Sequence-to-sequence models have also been investigated to directly generate ranking labels (Nogueira et al., 2020), and further training with explanations can yield improvements under lower-resource scenarios (Ferraretto et al., 2023). More recently, Sun et al. (2023) explored using the proprietary and exceptionally larger ChatGPT models for re-ranking[5]. Departing from existing ensembling techniques for re-ranking such as fusing bi-encoder embeddings (Lu et al., 2021), our method establishes the combination of discriminative and generative re-rankers through PMI maximization.

MI-based objectives, originally introduced in speech recognition to measure input-output dependence (Bahl et al., 1986; Woodland and Povey, 2002), have been applied to different tasks such as dialogue (Li et al., 2016), machine translation (Li and Jurafsky, 2016), and QA (Luo et al., 2022). MI-based joint inference and learning have been explored in question answering and generation (Tang et al., 2017), language understanding and generation (Su et al., 2020), and various vision and language tasks (Xia et al., 2017).

5 Conclusion

In this study, we introduce a simple and effective approach to enhance re-ranking for passage retrieval. By jointly utilizing a conventional cross-encoding re-ranker and a conditional query generator for inference, we optimize the pointwise mutual information between the query and passage distributions, achieving improvements in open-domain QA retrieval, and more significantly in zero-shot information retrieval tasks.

Acknowledgements

This research was supported by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd. under the Innovation and Technology Commission's InnoHK Scheme.

Limitations

First, improvements under the supervised setting for open-domain QA retrieval are diminished as K increases, and roughly equals out with using conventional re-rankers at $K = 20$; however, there are still many use cases especially for large models with limited context that can benefit from the improvements of our approach. Additionally, in this work we tackle passage re-ranking for retrieval, focusing on the second stage re-ranking scores using dense cross-encoders and generative models. We have not explored approaching the retrieval process without passage re-ranking, that is, directly applying the PMI objective to train a dense retrieval model, which could potentially lead to larger improvements but comes with much higher computational costs. We leave this for future work.

Ethics Statement

In this work, we used publicly available models and datasets for training and evaluation, and did not collect data or any personal information. The trained models could however potentially be misused and pose ethical risks typical of large language models when deployed in real-world applications, if not thoroughly audited.

References

L. Bahl, P. Brown, P. de Souza, and R. Mercer. 1986. Maximum mutual information estimation of hidden markov model parameters for speech recognition. In *ICASSP '86. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 11, pages 49–52.

Alexander Bondarenko, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. 2020. Overview of touché 2020: Argument retrieval. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 384–395, Cham. Springer International Publishing.

[5] Sun et al. (2023) reported results only on a subset of BEIR and uses BM25 "flat" (*cf.* "multifield").

Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. 2016. A full-text learning to rank dataset for medical information retrieval. In *Advances in Information Retrieval*, pages 716–722, Cham. Springer International Publishing.

Danqi Chen and Wen-tau Yih. 2020. Open-domain question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 34–37, Online. Association for Computational Linguistics.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Thomas Diggelmann, Jordan Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. 2020. Climate-fever: A dataset for verification of real-world climate claims. *arXiv preprint arXiv:2012.00614*.

Fernando Ferraretto, Thiago Laitz, Roberto Lotufo, and Rodrigo Nogueira. 2023. Exaranker: Synthetic explanations improve neural rankers. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2409–2414, New York, NY, USA. Association for Computing Machinery.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 2017. Dbpedia-entity v2: A test collection for entity search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, page 1265–1268, New York, NY, USA. Association for Computing Machinery.

Doris Hoogeveen, Karin M. Verspoor, and Timothy Baldwin. 2015. Cqadupstack: A benchmark data set for community question-answering research. In *Proceedings of the 20th Australasian Document Computing Symposium (ADCS)*, ADCS '15, pages 3:1–3:8, New York, NY, USA. ACM.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

Jiwei Li and Dan Jurafsky. 2016. Mutual information and diverse decoding improve neural machine translation. *arXiv preprint arXiv:1601.00372*.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*, pages 2356–2362.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jing Lu, Gustavo Hernandez Abrego, Ji Ma, Jianmo Ni, and Yinfei Yang. 2021. Multi-stage training with improved negative contrast for neural passage retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6091–6103, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345.

Hongyin Luo, Shang-Wen Li, Mingye Gao, Seunghak Yu, and James Glass. 2022. Cooperative self-training of machine reading comprehension. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 244–257, Seattle, United States. Association for Computational Linguistics.

Zhuang Ma and Michael Collins. 2018. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3698–3707, Brussels, Belgium. Association for Computational Linguistics.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: Financial opinion mining and question answering. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, page 1941–1942, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Reader-guided passage reranking for open-domain question answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 344–350, Online. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2017. MS MARCO: A human-generated MAchine reading COmprehension dataset.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document ranking with a pretrained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Revanth Gangi Reddy, Vikas Yadav, Md Arafat Sultan, Martin Franz, Vittorio Castelli, Heng Ji, and Avirup Sil. 2021. Towards robust neural retrieval models with synthetic pre-training. *arXiv preprint arXiv:2104.07800*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Devendra Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. 2022. Improving passage retrieval with zero-shot question generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3781–3797, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

G. Salton, A. Wong, and C. S. Yang. 1975. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6138–6148, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shang-Yu Su, Yung-Sung Chuang, and Yun-Nung Chen. 2020. Dual inference for improving language understanding and generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4930–4936, Online. Association for Computational Linguistics.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14918–14937, Singapore. Association for Computational Linguistics.

Duyu Tang, Nan Duan, Tao Qin, Zhao Yan, and Ming Zhou. 2017. Question answering and question generation as dual tasks. *arXiv preprint arXiv:1706.02027*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Christophe Van Gysel and Maarten de Rijke. 2018. Pytrec_eval: An extremely fast python interface to trec_eval. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 873–876, New York, NY, USA. Association for Computing Machinery.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R. Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. 2021. Trec-covid: Constructing a pandemic information retrieval test collection. *SIGIR Forum*, 54(1).

Ellen M Voorhees et al. 1999. The trec-8 question answering track report. In *Trec*, volume 99, pages 77–82. Citeseer.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. 2018. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 241–251, Melbourne, Australia. Association for Computational Linguistics.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Yining Wang, Liwei Wang, Yuanzhi Li, Di He, and Tie-Yan Liu. 2013. A theoretical analysis of ndcg type ranking measures. In *Conference on learning theory*, pages 25–54. PMLR.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

P.C. Woodland and D. Povey. 2002. Large scale discriminative training of hidden markov models for speech recognition. *Computer Speech & Language*, 16(1):25–47.

Yingce Xia, Tao Qin, Wei Chen, Jiang Bian, Nenghai Yu, and Tie-Yan Liu. 2017. Dual supervised learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3789–3798.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

A Open-Domain QA Retrieval Datasets

We show the number of train/dev/test examples in NQ and TriviaQA in Table 3. Please refer to Kwiatkowski et al. (2019) and Joshi et al. (2017) for more details. Note that NQ is licensed under Apache License 2.0, which we follow, and TriviaQA does not provide dataset licenses.

Dataset	Train	Dev	Test
Natural Questions	58,880	8,757	3,610
TriviaQA	60,413	8,837	11,313

Table 3: Dataset splits for NQ and TriviaQA.

B Open-Domain QA Retrieval Training and Inference Details

B.1 Training

Generally, conventional cross-encoders are trained to minimize the negative likelihood $\mathcal{L}_{\text{retrieval}}(\phi) \triangleq -\mathbb{E}_{x,z\sim p(\boldsymbol{x},\boldsymbol{z})}\left[\log p_{\boldsymbol{\phi}}(z|x)\right]$, where $p_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ is usually calculated from the retrieval score of question-passage pairs, with the partition function approximated by a noise contrastive approach trained either with a classification or a ranking objective (Ma and Collins, 2018). We choose to fine-tune our cross-encoder, BERT-FT, using a 6-layer transformer model (Vaswani et al., 2017), which takes the concatenated input of a query and a passage, with the binary classification objective for noise contrastive learning (Mikolov et al., 2013). The 6-layer SBERT model `MiniLM-L-6-v2` we use was previously pre-trained on MS MARCO, which we fine-tune for 2 epochs using the top 32 passages from BM25 on the NQ/TriviaQA training set. We train with a batch size of 128, learning rate of 5e-5, linear warmup and decay with ratio of 0.1.

For training of T5-FT, we fine-tune with $\mathcal{L}_{\text{generation}}(\boldsymbol{\theta})$ using the `t5-base-lm-adapt` model, a 12-layer encoder-decoder configuration with 220M parameters initialized from T5-base v1.1 and trained for an additional 100k steps with an LM objective. It takes a ground truth passage as input with its corresponding query as the decoder target. Ground truth query-passage pairs from the training set was used to fine-tune the model for 2 epochs. We use a batch size of 64, learning rate of 5e-5, and linear warmup and decay ratio of 0.1. Hyper-parameters were chosen by performance on the dev set.

UPR uses the pre-trained T0-3B directly without any fine-tuning.

JPR uses BERT-FT and T5-FT, described earlier, directly during inference (see Sec. B.2 below). JPR-FT requires further fine-tuning, which we train for another epoch. Training hyperparameters were searched with the dev set, with one run for each hyperparameter setting, shown in Table 4. We report results for the model with the best-performing run on the dev set.

All models were trained with HuggingFace's Transformers library (Wolf et al., 2020), using the AdamW optimizer (Loshchilov and Hutter, 2018) with default parameters. The maximum sequence lengths for queries and passages were set to 128 and 512, respectively, for generative models. For the cross-encoding BERT-FT, we set the maximum concatenated length to be 512. Training was done with four Nvidia A6000 GPUs, with around 2.5 GPU hours per epoch, equating to around 250 GPU-hours in total.

Hyper-parameter	NQ		TriviaQA	
	BERT-FT	T5-FT	BERT-FT	T5-FT
learning rate	1e-5	2e-5	1e-5	1.5e-5
batch size	96	64	64	64
α	0.0005	0.0005	0.005	0.005

Table 4: Training hyperparameters for NQ and TriviaQA selected by performance on the dev set.

B.2 Inference

For the conventional cross-encoding re-ranker (BERT-FT), we re-rank with Eq. 1 by directly ranking the retrieval scores. When using BERT-FT in JPR, we approximate $\log p_{\boldsymbol{\phi}}(z|x)$ by taking SoftMax over the scores for the 100 retrieved passages. For generative re-rankers T5-FT and UPR, we follow Sachan et al. (2022) and estimate $\log p_{\boldsymbol{\theta}}(x|z)$ with length-normalized conditional likelihood of the output sequence followed by taking SoftMax over the passages. For JPR, the preceding two terms are weight-averaged according to Eq. 3.

C Results on Open-Domain QA Retrieval with Different Cross-encoding and Generative Model Pairs

We further show the efficacy of JPR on NQ by conducting additional evaluations on NQ with various model combinations. We experiment with BERT models of different sizes for the cross-encoders, and for generative models we chose T5 models of multiple models sizes. All cross-encoding models were previously pre-trained on MS MARCO, which we fine-tune on NQ, and the T5 models were fine-tuned on NQ, all following training procedures reported in Sec. B. For inference, we use $\lambda = 0.5$ and follow the inference steps outlined in Sec. B.2. The results are shown in Table 5.

From the results, notice that when T5-small is paired with `MiniLM-L-6` for JPR, it aligns with the performance of T5-base paired with `MiniLM-L-6`. This observation underscores that the additional parameters of T5-base may be superfluous in our application. When comparing JPR (`MiniLM-L-6` & T5-small) with the standalone BERT-base, which is in the same parameter ballpark, and the larger BERT-large, it's evident that the gains from JPR

Cross-encoder	Generative Model	#params	Top-1	Top-5	Top-10
TinyBERT	✗	4.4M	37.8	60.3	67.0
MiniLM-L-4	✗	19.2M	47.5	65.9	70.9
MiniLM-L-6 (BERT-FT)	✗	22.7M	49.4	66.4	71.4
BERT-base	✗	109.5M	49.2	66.0	70.8
BERT-large	✗	335.1M	49.8	67.5	71.7
✗	T5-tiny	15.6M	25.7	51.4	62.0
✗	T5-small	77.0M	30.7	57.1	65.2
✗	T5-base (T5-FT)	247.6M	34.4	59.7	66.9
MiniLM-L-6	T5-tiny	38.3M	49.6	67.0	71.6
MiniLM-L-6	T5-small	99.7M	50.4	67.3	71.7
MiniLM-L-6	T5-base	270.3M	50.4	67.3	71.8

Table 5: Top-K retrieval accuracy (%) on NQ for different model combinations with the proposed JPR.

are not solely attributable to model size.

D BEIR Benchmark

The BEIR benchmark contains 18 datasets from a variety of text retrieval tasks and domains, 14 of which are publicly available. In this work we evaluate baselines and our approach on the publicly available datasets in BEIR: TREC-COVID (Voorhees et al., 2021), NFCorpus (Boteva et al., 2016), NQ (Kwiatkowski et al., 2019), HotpotQA (Yang et al., 2018), FiQA-2018 (Maia et al., 2018), ArguAna (Wachsmuth et al., 2018), Touché-2020 (Bondarenko et al., 2020), CQADupStack (Hoogeveen et al., 2015), Quora[6], DBPedia (Hasibi et al., 2017), SCIDOCS (Cohan et al., 2020), FEVER (Thorne et al., 2018), Climate-FEVER (Diggelmann et al., 2020), and SciFact (Wadden et al., 2020). For details on dataset statistics, links, and licenses please refer to BEIR (Thakur et al., 2021). Note that datasets in BEIR that are under copyright were not used in this study, and 4 out of the 14 publicly available datasets do not report dataset licenses. We follow the intended uses for each dataset license.

E Zero-shot Retrieval Training and Inference Details

For BEIR, since the SBERT model was already pre-trained on MS MARCO, we directly use it for BERT-FT. On the other hand, T5-FT stills requires fine-tuning, which we train for 3 epochs on query-passage pairs in the training set, with batch size of 16 and learning rate of 5e-5 with no warmup. The inference process is the same as open-domain QA retrieval, described earlier in Sec. B.2, except for λ which we set to 0.5 for all tasks as the BEIR tasks are zero-shot and we do not have access to the validation sets.

[6] https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs