InRanker: Distilled Rankers for Zero-shot Information Retrieval

Thiago Soares Laitz[1,2][0000−0001−7205−2094], Konstantinos Papakostas[3][0000−0002−2096−2931], Roberto Lotufo[1,4][0000−0002−5652−0852], and Rodrigo Nogueira[1,2,3][0000−0002−2600−6035]

[1] School of Electrical and Computing Engineering, State University of Campinas (UNICAMP), Campinas, Brazil
[2] Maritaca AI, Brazil
[3] Zeta Alpha, Netherlands
[4] NeuralMind, Brazil

Abstract. Despite multi-billion parameter neural rankers being common components of state-of-the-art information retrieval pipelines, they are rarely used in production due to the enormous amount of compute required for inference. In this work, we propose a method for distilling large rankers into their smaller versions focusing on out-of-domain effectiveness. We introduce InRanker, a version of monoT5 [25] distilled from monoT5-3B with increased effectiveness on out-of-domain scenarios. Our key insight is to use language models and rerankers to generate as much as possible synthetic "in-domain" training data, i.e., data that closely resembles the data that will be seen at retrieval time. The pipeline consists of two distillation phases that do not require additional user queries or manual annotations: (1) training on existing supervised soft teacher labels, and (2) training on teacher soft labels for synthetic queries generated using a large language model. Consequently, models like monoT5-60M and monoT5-220M improved their effectiveness by using the teacher's knowledge, despite being 50x and 13x smaller, respectively. Furthermore, we show that it is possible to transfer knowledge from English models to Portuguese fine-tuned models. Models and code are available at `https://github.com/unicamp-dl/inranker`

Keywords: Deep Learning · Language Models · Information Retrieval.

1 Introduction

It is well known that the effectiveness of IR pipelines increases with larger models [2,22,24,25,28]. For instance, multi-billion parameter rankers and dense models achieve top positions on leaderboards of IR benchmarks and competitions [10,11,12]. These large models leverage increased representation capacity, enabling them to encode features that might elude smaller models. However, deploying these large models is not without its challenges. The computational costs are substantial, often requiring specialized hardware such as GPUs or TPUs to operate in latency-critical applications. The high cost is directly related to the

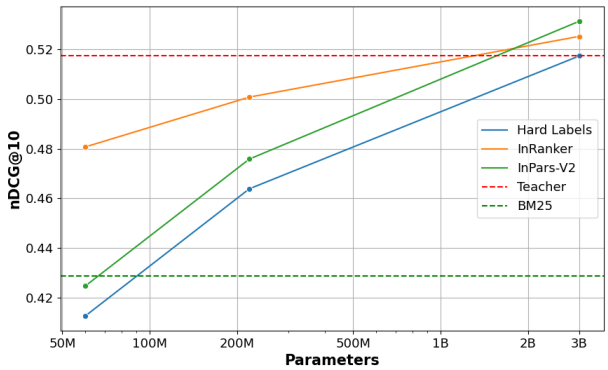

Fig. 1. Effectiveness on the BEIR benchmark [31]. All models are based on monoT5 [25], applying different fine-tuning methods.

large number of parameters that these models contain, as they require hardware with high memory and compute capacity and because the latency scales almost linearly with the number of parameters. In a production environment, this means higher operating costs and reduced scalability.

To address these challenges, there have been efforts to create more efficient models without significantly reducing effectiveness. One such approach is model distillation [17]. Distilled models, such as MiniLM [32], use a teacher or an ensemble of larger models to transfer knowledge to a smaller student model. Rosa et al.[30] show that MiniLM surpassed the zero-shot effectiveness of monoT5-base, which is a seq2seq model trained for binary classification, in IR tasks despite being an order of magnitude smaller in size. This has shown that knowledge transfer via model distillation is not only feasible but also effective. However, most distillation techniques have been geared towards optimizing effectiveness on specific benchmark tasks and do not focus on out-of-domain effectiveness. Rosa et al. also show that while smaller models are capable of achieving high in-domain results, similar to their larger counterparts, the disparity in effectiveness becomes evident in out-of-domain scenarios. As the concept of out-of-domain is subjective, we define it as a test distribution that is significantly different from the training distribution. A straighforward example of a out-of-domain scenario is when a model is trained on chemistry-related data and tested on legal data. However, we recognize that this distinction blurs in many scenarios.

Usually, training a retrieval model requires human-annotated hard labels informing which passage is relevant for each query. However, with the advance of Large Language Models (LLMs), it has become possible to generate synthetic queries for passages, providing a feasible approach for data augmentation [2,19,4,26,1]. Our work introduces a method for the generation of synthetic data specifically designed for distilling rankers that increases their out-of-domain effectiveness. We present InRanker, a distilled model derived from monoT5-3B [25], that uses the predictions of the teacher directly with both synthetic,

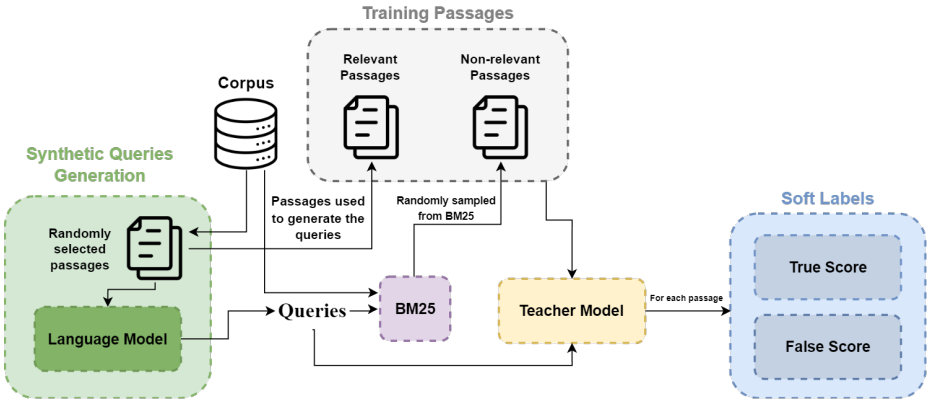

Fig. 2. Pipeline for generating the synthetic triples <query, passage, soft label> for the InRanker model.

generated from an out-of-domain corpus, and real query-document pairs. Effectively, this approach converts any corpus to be in-domain, since the model will be trained using queries from the target domain. As a result, this approach leads to reduced model sizes while maintaining improved out-of-domain effectiveness as presented in Figure 1. The methodology, results, and ablation experiments are presented in detail in the following sections.

2 Related Work

The research community has been using LLMs in a variety of tasks aimed at increasing the availability of data and improving the effectiveness of existing systems. Magister et al. [20] employed synthetic text generated by PaLM 540B [8] and GPT-3 175B [5] to transfer knowledge to smaller models such as T5. Fu et al. [15] successfully specialized student models in multi-step reasoning using FlanT5 [9] and code-davinci-002 as teachers. However, all these works rely on training the student models using synthetic text rather than directly using the soft labels. Furthermore, Muhamed et al. [21] distilled cross-attention scores of a language model for click-through-rate prediction, achieving better results when exposed to contextual features such as tabular data. Wang et al. [32] distilled the self-attention module, which is a crucial part of transformers, and successfully transferred knowledge to a variety of tasks.

Previous studies have also explored training a student from soft labels produced by a teacher: Hofstätter et al. [18] proposed a cross-architecture knowledge distillation approach using the MarginMSE loss. Similarly, Formal et al. [14] used the MarginMSE loss to distill knowledge to sparse neural models. Finally, Hashemi et al. [16] proposed a method for generating synthetic data for domain adaptation of dense passage retrievers. This approach involves creating new queries and a target collection, along with pseudo-labels extracted using

a BERT cross-encoder. However, they did not evaluate the model's effectiveness on datasets to which it was not domain-adapted. The existing research has mainly focused on in-domain evaluation, where the goal has been to increase the effectiveness of the student model on test datasets whose domain is similar to the datasets it was trained on. Our study also focuses on the robustness of the student and its ability to perform well even in out-of-domain scenarios, similar to the abilities of the larger teacher model.

3 Methodology

Our proposed method consists of two key phases of distillation, each designed with specific objectives to maximize the model's zero-shot effectiveness. The first phase uses real-world data to familiarize the student model with the ranking task, while the second phase uses synthetic data designed to improve zero-shot generalization and improve the model's effectiveness on a specific dataset. The dataset used to distill InRanker consists of {query, passage, logits} triplets, where the logits (soft labels) originate from a teacher model that has been trained for the relevance task. For the first stage, we chose to use query-document pairs from the MS MARCO [23] dataset, given their variety, the large number of annotated pairs, and its demonstrated effectiveness in enhancing retrieval effectiveness [29]. Next, we source the synthetic queries from InPars [2], which used an LLM to generate queries for the datasets in BEIR in a few-shot manner.

Distilling rerankers involves using the Mean Squared Error (MSE) loss to match the logits of the teacher and the student, as part of a two-phase pipeline illustrated in Figure 2. The first phase consists of two steps: (1) generating the teacher logits given a query and either a positive (relevant) or a negative (non-relevant) passage, where the negatives are randomly sampled using BM25 on the top-k = 1000 candidates, and the positives are sampled from the human-annotated pairs; and (2) training InRanker given the queries and passages as input using the MSE loss to match the student logits to those of the teacher, who remains frozen during training. This approach can be beneficial as it removes the need for making hard decisions about a passage's relevance, i.e. determining a threshold to obtain binary relevance labels, and instead focuses on a soft target objective aimed at aligning the student's perception of relevance with that of the teacher.

The second phase, with a focus on zero-shot effectiveness, uses the same two steps. However, instead of employing real queries sourced from a costly human-annotation process, it uses synthetic queries generated by an LLM based on randomly sampled documents from the corpus. In this scenario, the positive document is the one used to create the query, and the negatives are collected using the same top-k sampling approach as before.

We also perform zero-mean normalization on the teacher logits for each query-document pair, independent of the overall dataset distribution. This approach intends to make the data distribution symmetric for each query-document

pair, thereby minimizing the bias that InRanker is required to learn. Formally:

$$\begin{aligned} L'_{\text{true}} &= L_{\text{true}} - \frac{L_{\text{true}} + L_{\text{false}}}{2} \\ L'_{\text{false}} &= L_{\text{false}} - \frac{L_{\text{true}} + L_{\text{false}}}{2} \end{aligned} \tag{1}$$

with L_{true} and L_{false} denoting the teacher's logits for the relevant and non-relevant classes, respectively, and L' being the normalized values. This results in the following loss for each training example:

$$\mathcal{L}_{\text{MSE}} = ([Y_{\text{true}} - L'_{\text{true}}]^2 + [Y_{\text{false}} - L'_{\text{false}}]^2) \tag{2}$$

with Y_{true} and Y_{false} representing the logits of the student.

Due to the training objective described in equation (2), the model no longer determines the relevance of passages and instead focuses on replicating the teacher's output, thus eliminating the need for tunning a relevance threshold that would be needed to produce a binary label. With this approach, we can easily expand the out-of-domain knowledge of distilled models by generating new queries for documents using an LLM and fine-tuning the distilled model using the teacher's logits. In the experiments section, we demonstrate the effectiveness of this approach in enhancing the student model's effectiveness across 16 datasets of BEIR simultaneously. We present the hyperparameters used for training and the dataset curation in Appendix A, and we discuss variations of the training loss in Appendix C.

4 Experiments

4.1 English Knowledge Distillation Results

We distilled monoT5-3B to models with parameters ranging from 60M to 3B, using combinations of the following configurations:

Human Hard: representing the common approach for training rankers with human-annotated hard (i.e., binary) labels from the MS MARCO passage ranking dataset. In this case, a vanilla cross-entropy loss is used:

$$\mathcal{L}_{\text{CE}} = -\log P_{\text{relevant}} - \log P_{\text{non-relevant}} \tag{3}$$

where P_{relevant} and $P_{\text{non-relevant}}$ are the probabilities assigned by the model to the relevant and non-relevant query-document pair, respectively. Non-relevant pairs are sampled from the top-1000 retrieved by BM25.

Human Soft: representing a distillation step for matching the logits of a teacher and a student model, using real (human-generated) queries from the ranking dataset as inputs, but without the binary relevance judgments for targets.

Synthetic Soft: representing a distillation step for matching the logits of the two models, similar to the previous configuration, but using exclusively synthetic queries generated from the corresponding BEIR corpora with InPars [2,19].

Table 1. Distillation results (nDCG@10) on 16 BEIR datasets. The model marked with * represents the teacher model. We did not train InRanker-3B on human soft labels due to computational constraints.

Model	Training Configurations			Avg. Score
	Human Hard	Human Soft	Synthetic Soft	
(1) monoT5-60M	✓			0.4125
(2) ↪ w/ soft human		✓		0.4356
(3) InRanker-60M		✓	✓	**0.4807**
(4) monoT5-220M	✓			0.4638
(5) ↪ w/ soft human		✓		0.4870
(6) InRanker-220M		✓	✓	**0.5008**
(7) monoT5-3B*	✓			0.5174
(8) InRanker-3B	✓		✓	**0.5253**

From Table 1, we see that both distillation steps were essential for improving the average nDCG@10 score compared to the model trained solely using human hard labels from MS MARCO.[5] As a result, InRanker-60M (row 3) and InRanker-220M (row 6), despite being 50x and 13x smaller than the teacher model, were able to improve their effectiveness on the BEIR benchmark significantly. Moreover, models trained exclusively on MS MARCO soft labels (rows 2 & 5) saw an increase in effectiveness in comparison to training on solely hard labels (rows 1 & 4), corroborating findings from previous studies regarding the effectiveness of soft labels [17,18,14,16]. Furthermore, we observed an increase in the effectiveness even in self-distillation training (row 8), where the student learns soft labels generated by itself. We hypothesize that the improvement stems from the extra knowledge provided by the language model used to generate the synthetic queries. We did not provide results for the 3B model trained on both human soft and synthetic soft due to computational costs.

Furthermore, in Table 2, we present a effectiveness comparison between InRanker, Promptagator [13], and RankT5 [33]. Although we used monoT5-3B as a teacher for our experiments, which has a lower effectiveness on average when compared to Promptagator or RankT5-3B, our method is model-agnostic and thus one could use a stronger teacher model and anticipate even stronger results. Nonetheless, InRanker remains competitive in both model groups of 220M and 3B parameters, outperforming the other two baselines in 6 out of the 10 evaluated datasets, despite the average score not reflecting this due to Promptagator and RankT5 attaining a significantly higher score in two datasets: ArguAna and Touché.

[5] Results per dataset are shown in Appendix D.

Table 2. Comparison of the effectiveness for various reranking models, measured by nDCG@10 on the BEIR benchmark. The model marked with * represents the teacher model used for training InRanker. Bolded scores correspond to the best effectiveness on a specific dataset for a given model size, while underlined scores indicate the best effectiveness overall.

Dataset	InRanker	InRanker	Promptagator	RankT5-Enc	InRanker	monoT5	RankT5-Enc
	60M	220M	110M + 110M	220M	3B	3B*	3B
TREC-COVID	0.7775	**0.7984**	0.7620	0.7896	0.8175	0.7936	**0.8237**
NFCorpus	0.3547	0.3658	0.3700	**0.3731**	0.3825	0.3801	**0.3990**
HotpotQA	0.7563	**0.7742**	0.7360	0.7269	**0.7800**	0.7595	0.7536
Climate-FEVER	0.2729	**0.2914**	0.2030	0.2462	**0.2931**	0.2835	0.2753
DBPedia	0.4451	**0.4650**	0.4340	0.4373	**0.4762**	0.4719	0.4598
ArguAna	0.2466	0.2873	**0.6300**	0.3094	**0.4243**	0.3824	0.4069
Touché-2020	0.2883	0.2897	0.3810	**0.4449**	0.2924	0.3026	**0.4869**
SCIDOCS	0.1788	0.1911	**0.2010**	0.1760	**0.1990**	0.1978	0.1918
SciFact	0.7490	**0.7618**	0.7310	0.7493	**0.7831**	0.7773	0.7600
FiQA-2018	0.4043	0.4431	**0.4940**	0.4132	0.5027	**0.5068**	0.4932
Average	0.4474	0.4668	**0.4942**	0.4666	0.4951	0.4856	**0.5050**

4.2 Portuguese Knowledge Distillation Results

To further assess the efficacy of the technique in different languages, we evaluated InRanker on a Portuguese dataset for information retrieval: QUATI [6]. Instead of using the same T5 model, we started from PTT5, a Portuguese fine-tuned version of T5 [7,27]. We used the same two-step training approach as before, but with a strategy that allowed us to distill a Portuguese model using an English teacher (monoT5-3B). Given the availability of a translated version of MS MARCO in Portuguese [3], the first step (human soft) involved training the model using the Portuguese text, while matching the soft labels generated by the teacher using the original English text. This approach enabled us to leverage a stronger model that is not available in Portuguese for the distillation process. In the second step, involving synthetic soft labels from BEIR, we trained using the English text, as there is no translated version of BEIR available in Portuguese.

The results of this evaluation are presented in Table 3. We conclude that, similarly to English, the distillation process was able to improve the effectiveness of models in a zero-shot manner, as the models were trained using only real data from MS MARCO and synthetic data from BEIR. Remarkably, InRanker-740M surpassed the effectiveness of the mT5-3.7B on QUATI. Note that for the QUATI evaluation we used the same prompt presented in the paper to annotate all unjudged documents using gpt-4-turbo. Therefore all results are presented with a jugded@10 of 100%. Further results using synthetic data from QUATI and mixed training data are presented in Appendix E.

4.3 Ablations

In this section, we present our ablation experiments aimed at validating the best configuration for distilling monoT5-3B into smaller T5-based models, as well

Table 3. InRanker results on QUATI, a Portuguese evaluation dataset for information retrieval using PTT5-v2 [27]. All synthetic soft labels were generated using the BEIR datasets.

Model	Training Configurations			Avg. Score
	Human Hard	Human Soft	Synthetic Soft	
(1) PTT5-v2-60M	✓			0.4225
(2) ↪ w/ soft human		✓		0.4372
(3) InRanker-60M		✓	✓	**0.5121**
(4) PTT5-v2-220M	✓			0.5662
(5) ↪ w/ soft human		✓		0.5693
(6) InRanker-220M		✓	✓	**0.6108**
(7) PTT5-v2-740M	✓			0.5917
(8) ↪ w/ soft human		✓		0.6362
(9) InRanker-740M		✓	✓	**0.6624**
(10) monoT5-3B	✓			0.4864
(11) mT5-3.7B	✓			**0.6593**

as assessing their zero-shot capabilities. The initial experiments we conducted focused on evaluating how distillation would affect the model's effectiveness on novel dataset distributions that were not seen during training, i.e., we did not generate synthetic queries for them. To achieve this, we created two subsets, each containing 8 randomly selected datasets from 16 datasets of BEIR[6], which we named sample sets 1 and 2 and used only one set for training per experiment. The datasets that were used for training are designated as the "in-domain" category, while the remaining datasets, i.e. the other 8 datasets that are not part of the training set, represent the "out-of-domain" (O.O.D.) category.

Impact of soft knowledge distillation on O.O.D. effectiveness Our first ablation experiment focused on evaluating the initial distillation process using the MS MARCO dataset with soft labels. To accomplish this, we generated logits with monoT5-3B and trained both T5-base and T5-small models for 10 epochs. As shown in Table 4, rows 1-2 & 5-6, both models demonstrated an improvement in their nDCG@10 scores compared to the baseline, which was trained using the hard labels from MS MARCO. Remarkably, the overall score increased in both scenarios, even though the models were not exposed to any BEIR passages during this phase.[7]

Adding soft synthetic targets as a second distillation phase For the next experiment, we applied a second distillation step with synthetic soft labels on top of the model that we acquired from the last phase (monoT5 w/ soft human

[6] We list their exact composition in Appendix B.
[7] The individual results for each dataset are presented in Appendix D.

Table 4. Comparison of the in-domain vs out-of-domain effectiveness of our method, measured by nDCG@10. The model marked with * represents the teacher model used for the knowledge distillation process.

T5 Model	Training Configurations			Sample Set 1		Sample Set 2	
	Human Hard	Human Soft	Synthetic Soft	In-domain	O.O.D.	In-domain	O.O.D.
(1) 60M (monoT5)	✓			0.4141	0.4109	0.4817	0.3434
(2) 60M		✓		0.4422	0.4290	0.5124	0.3587
(3) 60M (InRanker)		✓	✓	**0.4768**	**0.4716**	**0.5558**	**0.3852**
(4) 60M	✓		✓	0.4475	0.4587	0.5355	0.3617
(5) 220M (monoT5)	✓			0.4647	0.4629	0.5475	0.3801
(6) 220M		✓		0.4867	0.4873	0.5692	0.4048
(7) 220M (InRanker)		✓	✓	**0.4945**	**0.5028**	**0.5874**	**0.4083**
(8) 220M	✓		✓	0.4905	0.4942	0.5832	0.3941
(9) 3B* (monoT5)	✓			0.5095	0.5253	0.6053	0.4295

labels). For that, we used the 100K synthetic queries generated by InPars for each dataset indicated as "in-domain" and trained for 10 epochs. As shown in Table 4, rows 3 & 7, while it was expected that the in-domain datasets would have an increase in their nDCG@10 scores, we observe that the out-of-domain datasets also had improvements, suggesting that the model's generalization capabilities were enhanced.

Using hard human targets for the first distillation phase Finally, we investigated the impact of skipping the first phase of distillation on MS MARCO logits, and instead starting from a model that was trained on hard human labels (monoT5-small and monoT5-base) and directly training using the synthetic soft BEIR targets. As we can see in Table 4, rows 3-4 & 7-8, when comparing with the model that was trained using the soft human targets, the overall effectiveness was reduced. From this, we conclude that the distillation step that includes the soft human targets on MS MARCO is beneficial, as it improves the model's effectiveness in both in-domain and out-of-domain scenarios.

Upper bound for soft distillation To estimate the upper bound of the effectiveness that these models could attain through distillation, we repeated the process using *real queries* from BEIR, (i.e., the validation queries) instead of the synthetic ones. Results presented in Table 5 show that for both model sizes, there was an increase in effectiveness for the in-domain datasets, as the model was exposed to the evaluation queries during training. However, we also observed an increase in effectiveness for out-of-domain datasets, indicating that the synthetic queries used for training could be improved.

Table 5. Upper bound effectiveness (nDCG@10) using real queries from BEIR for the distillation datasets. Bold indicates the best between using synthetic and real queries.

	Sample Set 1		**Sample Set 2**	
Model	In-domain	O.O.D.	In-domain	O.O.D.
InRanker-60M	0.4768	0.4716	0.5558	**0.3852**
↪ w/ real queries	**0.4975**	**0.4719**	**0.5860**	0.3813
InRanker-220M	0.4945	0.5028	0.5874	0.4083
↪ w/ real queries	**0.5242**	**0.5175**	**0.6159**	**0.4202**

5 Conclusion

This paper introduces a method for distilling the knowledge of information retrieval models and improve upon previous work how to better use synthetic data, aimed at improving the out-of-domain effectiveness of students. The study reveals that, through this knowledge distillation process, smaller models can achieve results comparable to the teacher, even in the context of multilingual transfer knowledge. This approach is particularly significant for applications where computational resources are limited, in production environments, or for languages with a lack of available models that can serve as a teacher. The methodology involves two steps of distillation: (1) using a human-curated corpus, and (2) using synthetic data generated by an LLM. Consequently, our work shows that it is possible to improve a reranker's capabilities in specific domains without requiring additional human-annotated labels. Finally, we observe that synthetic query generation could be improved since the real queries achieved a better out-of-domain effectiveness compared to the model trained solely on synthetic ones. However, the presented method has limitations. Specifically, it is not clear how to adapt the proposed loss to train dense retrievers, which tipically use a contrastive loss.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Alaofi, M., Gallagher, L., Sanderson, M., Scholer, F., Thomas, P.: Can generative llms create query variants for test collections? an exploratory study. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. p. 1869–1873. SIGIR '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3539618.3591960, https://doi.org/10.1145/3539618.3591960
2. Bonifacio, L., Abonizio, H., Fadaee, M., Nogueira, R.: InPars: Unsupervised Dataset Generation for Information Retrieval. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Informa-

tion Retrieval. pp. 2387–2392. SIGIR '22, Association for Computing Machinery, New York, NY, USA (2022). https://doi.org/10.1145/3477495.3531863, https://doi.org/10.1145/3477495.3531863
3. Bonifacio, L., Campiotti, I., de Alencar Lotufo, R., Nogueira, R.F.: mmarco: A multilingual version of MS MARCO passage ranking dataset. CoRR **abs/2108.13897** (2021), https://arxiv.org/abs/2108.13897
4. Boytsov, L., Patel, P., Sourabh, V., Nisar, R., Kundu, S., Ramanathan, R., Nyberg, E.: Inpars-light: Cost-effective unsupervised training of efficient rankers (2023)
5. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language Models are Few-Shot Learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 1877–1901. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf
6. Bueno, M., de Oliveira, E.S., Nogueira, R., Lotufo, R.A., Pereira, J.A.: Quati: A brazilian portuguese information retrieval dataset from native speakers (2024)
7. Carmo, D., Piau, M., Campiotti, I., Nogueira, R., Lotufo, R.: Ptt5: Pretraining and validating the t5 model on brazilian portuguese data (2020)
8. Chowdhery, A., Narang, S., Devlin, J., et. al: Palm: Scaling language modeling with pathways (2022)
9. Chung, H.W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., Webson, A., Gu, S.S., Dai, Z., Suzgun, M., Chen, X., Chowdhery, A., Castro-Ros, A., Pellat, M., Robinson, K., Valter, D., Narang, S., Mishra, G., Yu, A., Zhao, V., Huang, Y., Dai, A., Yu, H., Petrov, S., Chi, E.H., Dean, J., Devlin, J., Roberts, A., Zhou, D., Le, Q.V., Wei, J.: Scaling instruction-finetuned language models (2022)
10. Craswell, N., Mitra, B., Yilmaz, E., Campos, D.: Overview of the trec 2020 deep learning track (2021)
11. Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Lin, J.: Overview of the TREC 2021 deep learning track. In: Soboroff, I., Ellis, A. (eds.) Proceedings of the Thirtieth Text REtrieval Conference, TREC 2021, online, November 15-19, 2021. NIST Special Publication, vol. 500-335. National Institute of Standards and Technology (NIST) (2021), https://trec.nist.gov/pubs/trec30/papers/Overview-DL.pdf
12. Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Lin, J., Voorhees, E.M., Soboroff, I.: Overview of the TREC 2022 deep learning track. In: Soboroff, I., Ellis, A. (eds.) Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022. NIST Special Publication, vol. 500-338. National Institute of Standards and Technology (NIST) (2022), https://trec.nist.gov/pubs/trec31/papers/Overview_deep.pdf
13. Dai, Z., Zhao, V.Y., Ma, J., Luan, Y., Ni, J., Lu, J., Bakalov, A., Guu, K., Hall, K., Chang, M.W.: Promptagator: Few-shot Dense Retrieval From 8 Examples. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=gmL46YMpu2J
14. Formal, T., Lassance, C., Piwowarski, B., Clinchant, S.: From distillation to hard negative sampling: Making sparse neural ir models more effective. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. p. 2353–2359. SIGIR '22, Association for Comput-

ing Machinery, New York, NY, USA (2022). https://doi.org/10.1145/3477495.3531857, https://doi.org/10.1145/3477495.3531857
15. Fu, Y., Peng, H., Ou, L., Sabharwal, A., Khot, T.: Specializing smaller language models towards multi-step reasoning. In: Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J. (eds.) Proceedings of the 40th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 202, pp. 10421–10430. PMLR (23–29 Jul 2023), https://proceedings.mlr.press/v202/fu23d.html
16. Hashemi, H., Zhuang, Y., Kothur, S.S.R., Prasad, S., Meij, E., Croft, W.B.: Dense retrieval adaptation using target domain description. In: Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval. p. 95–104. ICTIR '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3578337.3605127, https://doi.org/10.1145/3578337.3605127
17. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network (2015)
18. Hofstätter, S., Althammer, S., Schröder, M., Sertkan, M., Hanbury, A.: Improving Efficient Neural Ranking Models with Cross-Architecture Knowledge Distillation (2020)
19. Jeronymo, V., Bonifacio, L., Abonizio, H., Fadaee, M., Lotufo, R., Zavrel, J., Nogueira, R.: Inpars-v2: Large language models as efficient dataset generators for information retrieval (2023)
20. Magister, L.C., Mallinson, J., Adamek, J., Malmi, E., Severyn, A.: Teaching small language models to reason. In: Rogers, A., Boyd-Graber, J., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 1773–1781. Association for Computational Linguistics, Toronto, Canada (Jul 2023). https://doi.org/10.18653/v1/2023.acl-short.151, https://aclanthology.org/2023.acl-short.151
21. Muhamed, A., Keivanloo, I., Perera, S., Mracek, J., Xu, Y., Cui, Q., Rajagopalan, S., Zeng, B., Chilimbi, T.: Ctr-bert: Cost-effective knowledge distillation for billion-parameter teacher models. In: NeurIPS Efficient Natural Language and Speech Processing Workshop (2021)
22. Neelakantan, A., Xu, T., Puri, R., Radford, A., Han, J.M., Tworek, J., Yuan, Q., Tezak, N., Kim, J.W., Hallacy, C., Heidecke, J., Shyam, P., Power, B., Nekoul, T.E., Sastry, G., Krueger, G., Schnurr, D., Such, F.P., Hsu, K., Thompson, M., Khan, T., Sherbakov, T., Jang, J., Welinder, P., Weng, L.: Text and code embeddings by contrastive pre-training (2022)
23. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: MS MARCO: A Human Generated MAchine Reading COmprehension Dataset (2016)
24. Ni, J., Qu, C., Lu, J., Dai, Z., Hernandez Abrego, G., Ma, J., Zhao, V., Luan, Y., Hall, K., Chang, M.W., Yang, Y.: Large dual encoders are generalizable retrievers. In: Goldberg, Y., Kozareva, Z., Zhang, Y. (eds.) Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. pp. 9844–9855. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Dec 2022). https://doi.org/10.18653/v1/2022.emnlp-main.669, https://aclanthology.org/2022.emnlp-main.669
25. Nogueira, R., Jiang, Z., Pradeep, R., Lin, J.: Document ranking with a pre-trained sequence-to-sequence model. In: Cohn, T., He, Y., Liu, Y. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2020. pp. 708–718. Association for Computational Linguistics, Online (Nov 2020). https://doi.

org/10.18653/v1/2020.findings-emnlp.63, https://aclanthology.org/2020.findings-emnlp.63
26. Penha, G., Palumbo, E., Aziz, M., Wang, A., Bouchard, H.: Improving content retrievability in search with controllable query generation. In: Proceedings of the ACM Web Conference 2023. p. 3182–3192. WWW '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3543507.3583261, https://doi.org/10.1145/3543507.3583261
27. Piau, M., Lotufo, R., Nogueira, R.: ptt5-v2: A closer look at continued pretraining of t5 models for the portuguese language (2024)
28. Pradeep, R., Nogueira, R., Lin, J.: The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models (2021)
29. Ren, R., Qu, Y., Liu, J., Zhao, X., Wu, Q., Ding, Y., Wu, H., Wang, H., Wen, J.R.: A thorough examination on zero-shot dense retrieval. In: Bouamor, H., Pino, J., Bali, K. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2023. pp. 15783–15796. Association for Computational Linguistics, Singapore (Dec 2023). https://doi.org/10.18653/v1/2023.findings-emnlp.1057, https://aclanthology.org/2023.findings-emnlp.1057
30. Rosa, G.M., Bonifacio, L., Jeronymo, V., Abonizio, H., Fadaee, M., Lotufo, R., Nogueira, R.: No parameter left behind: How distillation and model size affect zero-shot retrieval (2022)
31. Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., Gurevych, I.: 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) (2021), https://openreview.net/forum?id=wCu6T5xFjeJ
32. Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., Zhou, M.: Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 5776–5788. Curran Associates, Inc. (2020)
33. Zhuang, H., Qin, Z., Jagerman, R., Hui, K., Ma, J., Lu, J., Ni, J., Wang, X., Bendersky, M.: Rankt5: Fine-tuning t5 for text ranking with ranking losses. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. p. 2308–2313. SIGIR '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3539618.3592047, https://doi.org/10.1145/3539618.3592047

A Training Details

This appendix presents the parameters used for training the models using an A100 GPU with 80GB of VRAM. All experiments were conducted using the same learning rate of 7e-5 and the AdamW optimizer with its default hyperparameters in HuggingFace. The batch size was set to 32. For the 3B model, we used gradient checkpointing and gradient accumulation (to achieve an effective batch size of 2×16) due to memory constraints. During the generation of soft labels using the teacher model, we sampled 9 non-relevant passages passage for each relevant passage, leading to 10 pairs of logits per query. Differently from InPars and Promptagator, which train a separate model for each dataset, InRanker is a single model trained on all 16 datasets from BEIR.

B Datasets used in ablations

This section shows the datasets that were *randomly* chosen for inclusion in each sample set, resulting in the use of 12 out of the 16 BEIR datasets (as some were not used for training at all). Sample Set 1 includes the following datasets from the BEIR benchmark: NFCorpus, NQ, HotpotQA, DBPedia, Quora, SCIDOCS, FiQA-2018, and Signal-1M. Sample Set 2 comprises TREC-COVID, BioASQ, NQ, HotpotQA, Robust04, SCIDOCS, SciFact, and FiQA-2018. Climate-FEVER, TREC-NEWS, ArguAna, and Touché-2020 are not included in either sample set.

C Loss Function Ablation

We tested different loss functions, including the KL divergence and MSE, to match the logits of the two models. The results indicates that KL divergence was slightly worse for T5-small and that using only the true label in MSE as opposed to using both true and false labels also reduced the effectiveness. For a model with 60M parameters, the MSE with normalized logits yielded a result of 0.4807, while using the true logits only resulted in 0.4748. The KL divergence for the same model size was 0.4712. For a larger model with 220M parameters, the MSE with normalized logits produced a result of 0.5008, and the KL divergence was 0.5012. These results reflect the average nDCG@10 on 16 datasets of the BEIR benchmark with the varying loss functions.

D Complete Results on BEIR

Table 6 presents the results obtained after distilling the models using soft labels from MS MARCO and BEIR. We can observe the impact of both proposed distillation steps, namely using soft human labels and soft synthetic labels, which bring significant effectiveness improvements over the base models. In particular, using logits from MS MARCO leads to an average of a 2-point nDCG@10 improvement for each model, while the subsequent fine-tuning phase with the synthetic BEIR queries further enhances their effectiveness by 4.5 points for T5-small and approximately 1.4 points for T5-base.

E Complete Results on QUATI

Table 7 shows the results obtained by fine-tuning ptt5-v1 and ptt5-v2 [27] using different training sets. In particular, ptt5-v2 has a better overall nDGG@10, showing that even though both versions have the same number of parameters, a better pre-training process can improve downstream tasks such as information retrieval. For the QUATI soft labels generation we used the MT5-3.7B as a teacher instead of T5-3B, since this model has a better performance on Portuguese text.

Table 6. nDCG@10 values for each dataset after two steps of distillation.

Dataset	**T5-small** (60M)			**T5-base** (220M)			**T5-3B**
	Baseline	1st Step	+ 2nd Step	Baseline	1st Step	+ 2nd Step	Teacher
TREC-COVID	0.6928	0.7247	**0.7775**	0.7775	0.7643	**0.7984**	0.7936
NFCorpus	0.3180	0.3475	**0.3547**	0.3570	0.3639	**0.3658**	0.3801
BioASQ	0.4880	0.4648	**0.5516**	0.5240	0.5281	**0.5652**	0.5652
NQ	0.4733	0.5214	**0.5469**	0.5674	0.5855	**0.5971**	0.6251
HotpotQA	0.5996	0.6842	**0.7563**	0.6950	0.7546	**0.7742**	0.7595
Climate-FEVER	0.2116	0.2488	**0.2729**	0.2451	0.2739	**0.2914**	0.2835
DBPedia	0.3437	0.3745	**0.4451**	0.4195	0.4446	**0.4650**	0.4719
TREC-NEWS	0.3848	0.4478	**0.4646**	0.4475	**0.4808**	0.4695	0.4806
Robust04	0.4222	0.4782	**0.5386**	0.5016	0.5588	**0.5774**	0.6171
ArguAna	0.1274	0.1098	**0.2466**	0.1946	0.2431	**0.2873**	0.3824
Touché-2020	0.2643	0.2557	**0.2883**	0.2773	**0.2991**	0.2897	0.3026
Quora	0.8259	0.8246	**0.8335**	0.8230	0.8418	**0.8427**	0.8347
SCIDOCS	0.1436	0.1526	**0.1788**	0.1649	0.1746	**0.1911**	0.1978
SciFact	0.6963	0.7022	**0.7490**	0.7356	0.7505	**0.7618**	0.7773
FiQA-2018	0.3377	0.3712	**0.4043**	0.4136	0.4374	**0.4431**	0.5068
Signal-1M	0.2711	0.2612	**0.2820**	0.2771	0.2910	**0.2926**	0.3004
Average	0.4125	0.4356	**0.4807**	0.4638	0.4870	**0.5008**	**0.5174**

Table 7. Complete results (nDCG@10) on QUATI. All training datasets were used to distill knowledge from a teacher model.

	Training Datasets			**QUATI Evaluation**	
Parameters	MSMARCO	BEIR	QUATI	nDCG@10 ptt5-v1	nDCG@10 ptt5-v2
60M	✓			0.3838	0.4372
	✓	✓		0.4706	0.5121
	✓		✓	-	0.4993
	✓	✓	✓	0.4994	0.4999
220M	✓			0.5545	0.5693
	✓	✓		0.6129	0.6108
	✓		✓	-	0.5998
	✓	✓	✓	0.6270	0.6232
740M	✓			0.5818	0.6362
	✓	✓		0.6129	0.6624
	✓		✓	-	0.6468
	✓	✓	✓	0.6270	0.6570