Retrieval Capabilities of Large Language Models Scale with Pretraining FLOPs

Anonymous Author(s)
Affiliation
Address
email

Abstract

How does retrieval performance scale with pretraining FLOPs? We benchmark retrieval performance across LLM model sizes from 125 million parameters to 7 billion parameters pretrained on datasets ranging from 1 billion tokens to more than 2 trillion tokens. We find that retrieval performance on zero-shot BEIR tasks predictably scales with LLM size, training duration, and estimated FLOPs. We also show that In-Context Learning scores are strongly correlated with retrieval scores across retrieval tasks. Finally, we highlight the implications this has for the development of LLM-based retrievers.

1 Introduction

Industry labs as well as academic research groups have invested heavily in decoder-style LLMs. A consensus has grown around scaling laws for LLMs based on perplexity and downstream in context learning (ICL) tasks, where floating point operations (FLOPs) play an important role in addition to model size and training tokens. This has led to surprisingly capable LLMs with 7-8 billion active parameters or less (Touvron et al., 2023a,b; Jiang et al., 2023a; Dey et al., 2023; Dubey et al., 2024).

Recent work has shown that LLMs such as Llama 2 7B, Mistral 7B, and Mixtral 8x7B trained with causal language modeling can be naively converted into good retrieval models (Ma et al., 2023; Wang et al., 2023). In this study, we ask: **How well do decoder-style LLMs do on information retrieval tasks across model size, training duration, and FLOPs?**

The idea of taking pretrained checkpoints and converting them into good embedding models has been around since the early days of neural retrieval. For example, SentenceBERT successfully trained BERT models to embed sentence pairs (Reimers & Gurevych, 2019). Models such as GTR (Ni et al., 2021) and Instructor (Su et al., 2022) are initialized with the encoder weights from T5 models, and E5 is initialized with pretrained BERT weights (Wang et al., 2022).

In most of these cases, the models were further pretrained and/or finetuned with contrastive loss (a.k.a. InfoNCELoss), given the importunate fact that pretrained encoders and decoders are poor retrievers "out of the box." The approach of models like E5-base and E5-large (110M and 340M parameters respectively) is to continue pretraining on millions of query-passage pairs followed by finetuning on curated datasets such as MS MARCO (Bajaj et al., 2016). RepLlama on the other hand simply finetuned Llama 2 7B on 500,000 query-passage pairs from MS MARCO and found surprisingly good performance on zero-shot retrieval benchmarks such as BEIR (Ma et al., 2023; Thakur et al., 2021).

What is it about foundation models that allow them to perform well at information retrieval tasks? Given that foundation models like Llama 2 7B have been trained on 2 trillion tokens, can handle long context lengths and have strong in context learning capabilities, it makes sense that these new generation of LLMs do well on retrieval tasks. There has also been a recent trend where embedding

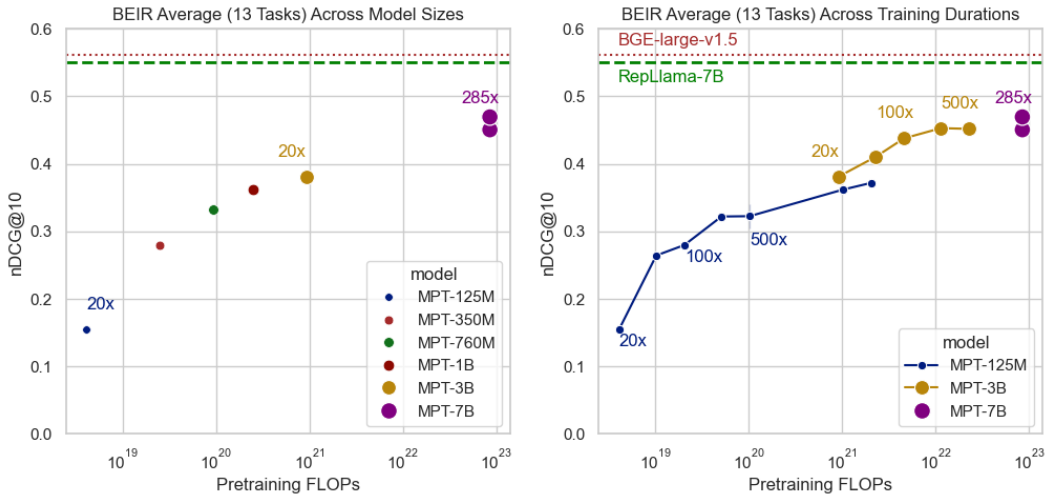

Figure 1: **Retrieval performance on BEIR improves with model size, training duration, and FLOPs.** (A) Increasing model size leads to an improvement in average BEIR nDCG@10 scores. MPT-125M, 350M, 760M, 1B, 3B models were trained for token parameter ratio of 20 on the same dataset, while MPT-7B models were trained for a token parameter ratio of 285 (i.e. a total of 2 trillion tokens). (B) Training models on more tokens leads to improvements in average BEIR score. Different MPT-125M models were trained for 3B to 1.5T tokens (20-10,000 token parameter ratios, blue line), while MPT-3B models were trained for 50B to 1.2T tokens (20-500 tokens per parameter, yellow line). All pretrained checkpoints were finetuned on 500k samples from MS MARCO with a maximum sequence length of 128 tokens.

models built on top of 7B parameter decoders have overtaken embedding models built on top of BERT-style models on benchmarks such as MTEB (Muennighoff et al., 2023b). Given the hard lesson that scale is (almost) all you need, **we set out to determine how retrieval performance is related to model size, training duration, and floating point operations (FLOPs)**.

In this study we explore the relationship between LLM model size, pretraining duration, and FLOPs for retrieval tasks. We start with pretrained checkpoints of MPT decoders (MosaicML NLP Team, 2023b; Sardana et al., 2023) ranging in size from 125 million parameters to 7 billion parameters. These checkpoints were trained on datasets ranging from 1 billion tokens to more than 2 trillion tokens spanning token to parameter ratios of 20 through 500 (and in one case 10,000). We minimally finetune each model checkpoint for one epoch of 500,000 MS MARCO samples with InfoNCE loss using contrastive pairs with hard negatives. We then analyze zero-shot retrieval performance on the BEIR benchmark Thakur et al. (2021). We find that:

- Retrieval performance scales both with increasing model size and increasing pretraining duration for a fixed model size. This relationship is best captured by accuracy-FLOPs curves; i.e. **retrieval performance scales with pretraining FLOPs**.
- For most of the BEIR retrieval tasks, training a small model for more tokens has similar accuracy to a larger model trained on fewer tokens up to a ceiling. Another way of stating this is that **isoFLOPs curves for fixed model sizes significantly overlap**.
- In Context Learning (ICL) scores are strongly correlated with retrieval scores across BEIR tasks. Almost without exception, LLMs that have higher ICL scores also have higher retrieval scores.

The goal of this study is not to achieve SOTA retrieval performance. Rather it is to investigate scaling properties as a function of tokens seen during pretraining, which has not been investigated in prior work. We believe this has important implications for the next generation of dense retrieval models; our results indicate that high quality decoders ranging in size from 1-8B active parameters are strong candidates for future embedding models.

2 Results

We used checkpoints of pretrained MPT decoders of sizes varying from 125 million parameters to 7 billion parameters. These checkpoints were trained for various durations ranging from 1 billion tokens to 2 trillion tokens with corresponding token-per-parameter ratios of 20 through 500 (and in one case up to 10,000). Note that each checkpoint was trained to completion, with a full learning rate schedule consisting of a warmup followed by a cosine decay. See MosaicML NLP Team (2023b,a) for further pretraining details. We then minimally finetuned these models on MS MARCO for 1 epoch (500,000 samples) of query-passage pairs with hard negatives and evaluated them on the BEIR retrieval benchmark.

2.1 Finetuning on MS MARCO with the InfoNCELoss

Using a pretrained LLM, we pool and average the respective tokens of the final hidden representations of each document and query.[1] To finetune the model, we leverage the class InfoNCE loss as follows over the in-batch negatives and hard negatives:

$$\min L_{\text{cont}} = -\frac{1}{n}\sum_i \log \frac{\mathrm{e}^{s_{\boldsymbol{\theta}}(q_i,p_i)}}{\mathrm{e}^{s_{\boldsymbol{\theta}}(q_i,p_i)} + \sum_j \mathrm{e}^{s_{\boldsymbol{\theta}}\left(q_i,p_{ij}^-\right)}} \tag{1}$$

where q_i is a query, p_i is a "positive" passage that is paired with the query, and p_{ij}^- is a "hard negative" passage that is somewhat relevant to the query but not the correct passage. s_θ is the cosine similarity function, and n is the total number of samples in a batch. This formula can be expressed in terms of cross entropy, which makes for easy implementation in PyTorch. Our implementation is motivated by Wang et al. (2022).[2]

The positive document for a given query is derived from the MS MARCO Document Retrieval dataset (Bajaj et al., 2016). We use 15 curated hard negative documents per query, mined using BM25 (Robertson et al., 1995).

While all MPT models were pretrained with a maximum sequence length of 4096 tokens, we finetuned and evaluated all models with a maximum sequence length of 128 tokens. We also used the same hyperparameters such as warmup and learning rate for all model sizes without doing hyperparameter sweeps. This strongly handicaps the models; we therefore interpret all our BEIR scores as *lower bounds* on retrieval performance.

2.2 Estimating FLOPs

Floating point operations (FLOPs) is a hardware-agnostic metric that conveys how much "compute" was used to train a particular model. To first order, FLOPs for dense transformer models can be estimated as $6 \times N \times tokens$ where N is the total number of model parameters (Kaplan et al., 2020; Chowdhery et al., 2023; Anthony et al., 2023). Thus FLOPs increase with both training duration (i.e. data) and model size. We use this approximation for all FLOPs estimates.

2.3 BEIR Retrieval Benchmark

We describe the MTEB Muennighoff et al. (2023b) version of BEIR (Thakur et al., 2021), which we use for all of our evaluations. Each benchmark within BEIR is divided into queries and documents (a.k.a "passages"), and the task is to find most relevant documents for a given query. Exact search is done using cosine similarity (as opposed to approximate search). SCIDOCS is one example dataset that contains 1000 Queries, and 25,657 Documents from scientific publications in the test set (Cohan et al., 2020). The various BEIR tasks are detailed in the Appendix D.

BEIR was originally designed to be a zero-shot evaluation benchmark, which means that many of the early retrieval models were careful *not* to train in-domain. However, all of the "top" embedding models on the MTEB benchmark not only train on the training sets associated with each BEIR

[1] Other approaches here include using an end of sentence token to represent the content of the preceding tokens.

[2] `https://github.com/microsoft/unilm`

benchmark, but also arguably train on the corpus test sets of each BEIR task (Wang et al., 2022, 2023; Xiao et al., 2023). Since many of the benchmarks are derived from common datasets derived from Wikipedia, Stack Exchange, many argue that it is unrealistic to treat BEIR as a zero-shot evaluation benchmark. Instead of focusing on benchmark hacking, in this study we choose to benchmark the scaling properties of pretrained decoders by minimally finetuning on a single embedding dataset (MS MARCO). Future work can use more curated, higher-quality finetuning datasets to show similar scaling properties.

2.4 Retrieval Performance Scales with Model Size, Training Duration, and FLOPs

We first evaluated MPT 125M, 350M, 760M, 1B and 3B models all pretrained to a token per parameter ratio of 20, ranging from roughly 3 billion tokens to 50 billion tokens (see Appendix Table 2). Figure 1A shows the smooth increase of average BEIR score with model size as a function of pretraining FLOPs.

We then used MPT-125M model checkpoints for various training durations ranging from 3 billion tokens, i.e. 20 tokens per parameter, through 1.5 trillion tokens, i.e. an extreme of 10,000 tokens per parameter (Figure 1B blue line). Somewhat surprisingly, *we find that retrieval performance steadily improves with increased training duration and does not plateau*. We also used pretrained MPT-3B model checkpoints for various training durations ranging from 50B i.e. 20 tokens per parameter, to 1.2 trillion tokens, i.e. 500 tokens per parameter. Here too we see a steady increase in retrieval performance as a function of FLOPs. Finally, we also finetune two slightly different MPT 7B checkpoints pretrained on 2T tokens with a token per parameter ratio of 285 (Figure 1 purple dots). As expected, these have slightly higher average nDCG@10 than the MPT-3B checkpoint trained to 500 tokens per parameter.

Figure S1 shows scores for individual BEIR tasks as a function of pretraining FLOPs. We show Llama 2 7B and Mistral 7B performance on some tasks for comparison (finetuned in the same manner as all of the other checkpoints).[3] As expected, the scaling trends for individual BEIR tasks are the same as in Figure 1, with the notable exceptions of Arguana and Touche 2020 (not shown). For Arguana, performance essentially plateaus despite increasing FLOPs. This is likely due to the unusual nature of the task, which requires the model to find a *counterargument* for a given argument text (i.e. the query). The only models that excel at this task are models that explicitly include *instructions to find the counterargument* in the query, like E5-Mistral-7B (Wang et al., 2023).[4] We speculate on the role of explicit instructions in the Discussion.

2.5 Small models trained on more data can match performance of larger models trained on less data

One surprising observation from Figure 1 and Figure S1 is that small models pretrained on more data (i.e. for more FLOPs) can match the performance of larger models trained on less data. For example, MPT-125M trained for 1.5T tokens at roughly a 10,000 token parameter ratio has same average BEIR performance as a MPT-1B model trained for 25.2B tokens with a token parameter ratio of 20.

2.6 BEIR scores are strongly correlated with ICL Scores

We then plot the BEIR nDCG@10 scores as a function of the ICL scores for each pretrained checkpoint using the open-source MosaicML Evaluation Gauntlet (Dohmann, 2023; Barton, 2024). Figure S2 demonstrates that ICL scores of pretrained checkpoints are strongly correlated with BEIR scores after contrastive finetuning on MS MARCO. This provides strong evidence that BEIR is a good measure of LLM retrieval capability. See Appendix S3 for the nDCG vs. ICL performance for individual BEIR tasks.

There is a noticeable gap in average ICL score between MPT-125M with token parameter ratio 10,000 (blue line) and MPT-3B with token per parameter ratio 20 (yellow) along the x-axis for all tasks. This simply indicates that MPT-3B models are always better than MPT-125M models for the ICL tasks.

[3] Note here that the Mistral 7B pretraining FLOPs are unknown; we simply assume that Mistral was trained on slightly more data than Llama 2 7B.

[4] See Appendix Table 14, where the evaluation instructions are "Given a claim, find documents that refute the claim"

References

Quentin Anthony, Stella Biderman, and Hailey Schoelkopf. Transformer math 101, 2023. URL blog.eleuther.ai.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. MS MARCO: A human generated machine reading comprehension dataset. *arXiv:1611.09268*, 2016. URL https://arxiv.org/abs/1611.09268.

Tessa Barton. Calibrating the mosaic evaluation gauntlet, April 2024. URL https://www.databricks.com/blog/calibrating-mosaic-evaluation-gauntlet.

Alexander Bondarenko, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, et al. Overview of touché 2020: argument retrieval. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, Proceedings 11*, pp. 384–395. Springer, 2020.

Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. A full-text learning to rank dataset for medical information retrieval. In *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20–23, 2016. Proceedings 38*, pp. 716–722. Springer, 2016.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Jianlyu Chen, Nan Wang, Chaofan Li, Bo Wang, Shitao Xiao, Han Xiao, Hao Liao, Defu Lian, and Zheng Liu. Air-bench: Automated heterogeneous information retrieval benchmark. *arXiv preprint arXiv:2412.13102*, 2024.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

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

Harm De Vries. Go smol or go home, 2023. URL https://www.harmdevries.com/post/model-size-vs-compute-overhead/.

Nolan Dey, Daria Soboleva, Faisal Al-Khateeb, Bowen Yang, Ribhu Pathria, Hemant Khachane, Shaheer Muhammad, Robert Myers, Jacob Robert Steeves, Natalia Vassilieva, et al. Btlm-3b-8k: 7b parameter performance in a 3b parameter model. *arXiv preprint arXiv:2309.11568*, 2023.

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

Jeremy Dohmann. Blazingly fast llm evaluation for in-context learning, February 2023. URL https://www.databricks.com/blog/llm-evaluation-for-icl.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yan Fang, Jingtao Zhan, Qingyao Ai, Jiaxin Mao, Weihang Su, Jia Chen, and Yiqun Liu. Scaling laws for dense retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1339–1349, 2024.

Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 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*, pp. 1265–1268, 2017.

Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*, 2021.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Doris Hoogeveen, Karin M Verspoor, and Timothy Baldwin. Cqadupstack: A benchmark data set for community question-answering research. In *Proceedings of the 20th Australasian document computing symposium*, pp. 1–8, 2015.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*, 2021.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023a.

Ting Jiang, Shaohan Huang, Zhongzhi Luan, Deqing Wang, and Fuzhen Zhuang. Scaling sentence embeddings with large language models. *arXiv preprint arXiv:2307.16645*, 2023b.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*, 2020.

Omar Khattab and Matei Zaharia. 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*, pp. 39–48, 2020.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*, 2024.

Jinhyuk Lee, Feiyang Chen, Sahil Dua, Daniel Cer, Madhuri Shanbhogue, Iftekhar Naim, Gustavo Hernández Ábrego, Zhe Li, Kaifeng Chen, Henrique Schechter Vera, et al. Gemini embedding: Generalizable embeddings from gemini. *arXiv preprint arXiv:2503.07891*, 2025.

Chaofan Li, MingHao Qin, Shitao Xiao, Jianlyu Chen, Kun Luo, Yingxia Shao, Defu Lian, and Zheng Liu. Making text embedders few-shot learners. *arXiv preprint arXiv:2409.15700*, 2024.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. Fine-tuning llama for multi-stage text retrieval. *arXiv preprint arXiv:2310.08319*, 2023.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. Www'18 open challenge: financial opinion mining and question answering. In *Companion proceedings of the the web conference 2018*, pp. 1941–1942, 2018.

Luke Merrick, Danmei Xu, Gaurav Nuti, and Daniel Campos. Arctic-embed: Scalable, efficient, and accurate text embedding models. *arXiv preprint arXiv:2405.05374*, 2024.

MosaicML NLP Team. Introducing mpt-30b: Raising the bar for open-source foundation models. `www.mosaicml.com/blog/mpt-30b`, 2023a. Accessed: 2023-06-22.

MosaicML NLP Team. Introducing mpt-7b: A new standard for open-source, commercially usable llms. `www.mosaicml.com/blog/mpt-7b`, 2023b. Accessed: 2023-05-05.

Niklas Muennighoff. Sgpt: Gpt sentence embeddings for semantic search. *arXiv preprint arXiv:2202.08904*, 2022.

Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models. *arXiv preprint arXiv:2305.16264*, 2023a.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2006–2029, 2023b.

Niklas Muennighoff, Hongjin Su, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. Generative representational instruction tuning. *arXiv preprint arXiv:2402.09906*, 2024.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y Zhao, Yi Luan, Keith B Hall, Ming-Wei Chang, et al. Large dual encoders are generalizable retrievers. *arXiv preprint arXiv:2112.07899*, 2021.

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

Zach Nussbaum, John X Morris, Brandon Duderstadt, and Andriy Mulyar. Nomic embed: Training a reproducible long context text embedder. *arXiv preprint arXiv:2402.01613*, 2024.

Jacob Portes, Alexander Trott, Sam Havens, Daniel King, Abhinav Venigalla, Moin Nadeem, Nikhil Sardana, Daya Khudia, and Jonathan Frankle. Mosaicbert: A bidirectional encoder optimized for fast pretraining. *Advances in Neural Information Processing Systems*, 36:3106–3130, 2023.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*, 2021.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. Okapi at trec-3. *Nist Special Publication Sp*, 109:109, 1995.

Shafiq Rayhan Joty Caiming Xiong Yingbo Zhou Semih Yavuz Rui Meng, Ye Liu. Sfr-embedding-mistral:enhance text retrieval with transfer learning. Salesforce AI Research Blog, 2024. URL `https://www.salesforce.com/blog/sfr-embedding/`.

Nikhil Sardana, Jacob Portes, Sasha Doubov, and Jonathan Frankle. Beyond chinchilla-optimal: Accounting for inference in language model scaling laws. *arXiv preprint arXiv:2401.00448*, 2023.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, and Tao Yu. One embedder, any task: Instruction-finetuned text embeddings. *arXiv preprint arXiv:2212.09741*, 2022.

Hongjin Su, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han-yu Wang, Haisu Liu, Quan Shi, Zachary S Siegel, Michael Tang, et al. Bright: A realistic and challenging benchmark for reasoning-intensive retrieval. *arXiv preprint arXiv:2407.12883*, 2024.

Yi Tay, Mostafa Dehghani, Jinfeng Rao, William Fedus, Samira Abnar, Hyung Won Chung, Sharan Narang, Dani Yogatama, Ashish Vaswani, and Donald Metzler. Scale efficiently: Insights from pre-training and fine-tuning transformers. *arXiv preprint arXiv:2109.10686*, 2021.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 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. URL `https://openreview.net/forum?id=wCu6T5xFjeJ`.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 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)*, pp. 809–819, 2018.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971, 2023a. URL `https://api.semanticscholar.org/CorpusID:257219404`.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Pablo Villalobos and David Atkinson. Trading off compute in training and inference, 2023. URL `https://epochai.org/blog/trading-off-compute-in-training-and-inference`. Accessed: 2024-02-01.

Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. Trec-covid: constructing a pandemic information retrieval test collection. In *ACM SIGIR Forum*, volume 54, pp. 1–12. ACM New York, NY, USA, 2021.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. 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)*, pp. 241–251, 2018.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. *arXiv preprint arXiv:2004.14974*, 2020.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*, 2023.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, et al. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*, 2024.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighof. C-pack: Packaged resources to advance general chinese embedding. *arXiv preprint arXiv:2309.07597*, 2023.

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

Hansi Zeng, Julian Killingback, and Hamed Zamani. Scaling sparse and dense retrieval in decoder-only llms. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2679–2684, 2025.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

Dawei Zhu, Liang Wang, Nan Yang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. Longembed: Extending embedding models for long context retrieval. *arXiv preprint arXiv:2404.12096*, 2024.

Model Name	Total Parameters	Hidden Dimension	Attention Heads	Layers	Expansion Ratio
MPT-125M	151M	768	12	12	4
MPT-350M	367M	1024	16	24	4
MPT-760M	749M	1536	12	24	4
MPT-1B	1.26B	2048	16	24	4
MPT-3B		2560	20	32	4
MPT-7B		4096	32	32	4

Table 1: MPT Model Architecture Details

TPR	MPT-125M	MPT-350M	MPT-760M	MPT-1B	MPT-3B	MPT-7B	Llama-7B
20	3.02B	7.34B	14.98B	25.2B	49.2B	-	-
50	7.55B	18.35B	-	63B	123B	-	-
100	15.1B	36.7B	74.9B	126B	246B	700B	-
250	37.8B	73.4B	187.25B	-	615B	-	-
285	-	-	-	-	-	2000B	2000B
500	75.5B	183.5	374.5B	-	1230B	-	-
1000	151B	-	-	-	-	-	-
5000	755B	-	-	-	-	-	-
10000	1510B	-	-	-	-	-	-

Table 2: Training Duration (tokens). Note that one MPT-125M checkpoint was trained for 1.5 trillion tokens.

A Related Work

A.1 Embedding/Dense Retrieval Models

The idea of taking pretrained LLM checkpoints and converting them into embedding models has been around since the early days of neural retrieval. Dense retrieval quickly grew in popularity after the release of BERT. For example, SentenceBERT successfully trained BERT models to embed sentence pairs (Reimers & Gurevych, 2019). Early approaches focused on re-ranking (Nogueira & Cho, 2019), although "full ranking" (i.e. embedding models) shortly followed (Khattab & Zaharia, 2020; Karpukhin et al., 2020; Izacard et al., 2021).

Ni et al. (2021) showed that increase LLM model size while keeping the *final embedding dimension fixed* led to improvements in zero-shot BEIR retreival performance. Specifically, in order to build their GTR models, they used the encoder half of T5-Base (110M), large (335M), XL (1.24B) and XXL (4.8B) and further pretrained them on 2 billion community question-answer pairs and then finetuned them on MS MARCO (Bajaj et al., 2016). Our work builds on this direction by using decoders instead of encoders and by avoiding the "further pretraining" stage altogether.

Many studies have shown that BERT-base and BERT-Large size models can achieve state-of-the-art performance on retrieval benchmarks such as BEIR when trained with contrastive pairs Wang et al. (2022); Xiao et al. (2023). Wang et al. (2022) trained E5 on millions of contrastive pairs with soft negatives but did not release their pretraining data, while with BGE, Xiao et al. (2023) followed a similar recipe and gave more hints as to their pretraining data sets.

Ma et al. (2023) finetuned Llama 7B weights on MS MARCO and found very good performance on BEIR Thakur et al. (2021). Our work builds on RepLlama by exploring the scaling properties of LLM-based retrieval models.

Some of the same authors as RepLlama and E5 achieved state of the art performance on BEIR by finetuning Mistral-7B-Instructor (Jiang et al., 2023a) on 13 datasets of contrastive pairs with hard negatives as well as synthetic contrastive pairs generated by GPT-3.5/4 Wang et al. (2023).

There has been a recent uptick in BERT-style embedding models including Nomic Embed Nussbaum et al. (2024), Arctic-Embed (Merrick et al., 2024), ModernBERT (Warner et al., 2024), Similarly, there has been a recent explosion of LLM-based embedding models such as SFR-Embedding-Mistral (Rui Meng, 2024), NV-Embed (Lee et al., 2024), bge-en-icl (Li et al., 2024), gte-Qwen2-1.5B-instruct

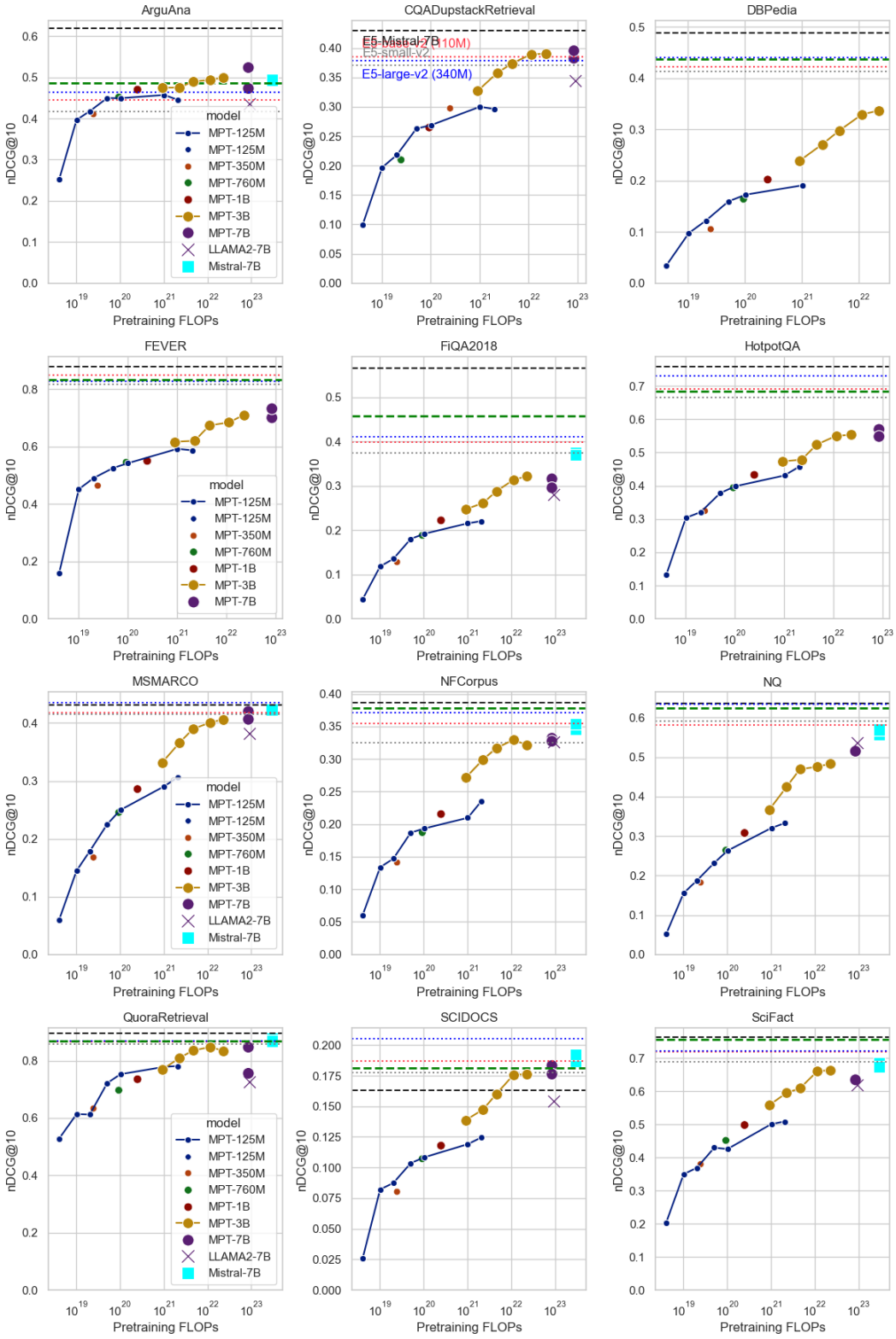

Figure S1: **LLM performance on individual BEIR tasks scales with model size, pretraining duration, and FLOPs**. E5-small-v2 (grey), E5-base-v2(red), E5-large-v2 (blue), E5-Mistral-7B (black), and RepLlama-7B (green) performance is included for reference.

(Li et al., 2023), GRIT LM (Muennighoff et al., 2024) and others. The Qwen 3 Embedding models (0.6B, 4B 8B) models were initialized from the Qwen 3 pretrained models (Zhang et al., 2025). Similarly, the Gemini Embedding models (Lee et al., 2025) were initialized from the Gemini models (Team et al., 2023).

A.2 Evaluation of Embedding Models

Neural network based embedding and retrieval were considered slightly different subfields. Over time these have converged, as exemplified by the MTEB benchmark (Muennighoff et al., 2023a), which incorporates BEIR (Thakur et al., 2021) as well as many other benchmarks for tasks such as semantic similarity. We give more details on the BEIR and MTEB benchmarks below. More recent retrieval benchmarks include AIR Bench Chen et al. (2024), LongEmbed (Zhu et al., 2024) and BRIGHT Su et al. (2024); we save the analysis of LLM scaling properties on these benchmarks for future work.

A.3 Scaling Laws for Large Language Models

The study by Hoffmann et al. (2022), informally known as the "Chinchilla paper," determined scaling laws for optimally allocating train compute for LLMs, resulting in a heuristic that models should be trained on a total number of tokens that is roughly $20\times$ the number of model parameters (i.e. the token-per-parameter ratio). They use three approaches to predict optimal model size and number of tokens for pretraining a LLM with a fixed compute budget; first they fix model size and vary dataset size, and then they also establish isoFLOP profiles by varying model size for fixed number of FLOPs. Finally, they fit a parametric loss function. All approaches arrive at the same rough conclusion that LLMs such as GPT-3 (Brown et al., 2020) and Gopher (Rae et al., 2021) were significantly under trained. This builds on previous work exploring scaling laws for LLMs by Hernandez et al. (2021); Kaplan et al. (2020); Tay et al. (2021).

The popularity - and capabilities - of models such as Llama 2 7B, Mistral 7B (Jiang et al., 2023a) and Llama 3 8B have upended the chinchilla scaling laws. These models were trained on far more tokens than "Chinchilla optimal;" for example, Llama 2 7B was pretrained for 2 trillion tokens, which is a token parameter ratio of 285. Recent studies have pointed to the fact that consideration such as inference serving cost (which increases with model size) as well as ease of use should be taken into consideration when training LLMs (De Vries, 2023; Sardana et al., 2023; Villalobos & Atkinson, 2023).

All these studies focus on cross entropy loss and not on downstream ICL performance (or retrieval performance for that matter). While a study by Tay et al. (2021) found that T5 model size did not follow easily identifiable scaling laws on downstream SuperGLUE performance, there is little work showing how current decoder downstream performance scales with FLOPs. There are even fewer studies that focus on the question of how LLM retrieval performance scales with FLOPs.

Muennighoff (2022) investigated the scaling properties of GPT for semantic similarity from 100M - 5.8B parameters. Motivated by the promise of LLM-based retrievers, Jiang et al. (2023b) propose an ICL-based method to improve sentence embedding performance and use it to finetune OPT models from size 125M to 66B and Llama models across 7B, 13B, 30B and 65B. In this work, however, they exclusively focus on the semantic textual similarity benchmarks. Finally, Fang et al. (2024) derived phenomenological scaling laws for dense embeddings based on BERT models up to 80M parameters. Zeng et al. (2025) investigate the retrieval across the Llama 3 1B, 3B and 8B models. Our study complements these results and focuses on the scaling properties of decoders from 125M parameters up to 7B parameters with varying token per parameter ratios.

B Discussion

Our work shows that retrieval performance scales with LLM pretraining FLOPs, and we hope that it provides strong motivation for finetuning pretrained LLMs for retrieval.

Why might one want to build an embedding model using a pretrained LLM? There has been a cambrian explosion of open-source LLMs in the range of 1B, 3B, 7B, 8B, and 13B parameters (as

well as Mixture of Experts models with a similar number of active parameters).[5] Almost all of these models have been pretrained on *trillions* of tokens, can handle long context lengths upwards of 8k tokens, and have been extensively finetuned to handle nuanced language (using both supervised finetuning and reinforcement learning).

The previous generation of embedding models such as E5, GTE and BGE were built on top of pretrained BERT models. BERT is 6 years old and counting, and there is much less development occurring for small encoder models.[6] While BERT-style models are smaller than 1B parameters and therefore easier to deploy, 7B models are already dominating retrieval benchmarks such as MTEB. We suspect that this trend away from BERT models will continue.

Does retrieval performance continue to increase beyond 7B models? While there are mixed reports on the success of retrieval models larger than 7B parameters (e.g. RepLlama 13B, Mixtral 8x22B), our results hint at the tantalizing possibility that retrieval performance might continue to scale beyond 7B parameters. With the recent state of the art open weights models such as DeepSeek, Qwen 3, and Llama 4, we expect research on decoder-based retrieval to continue in this direction.

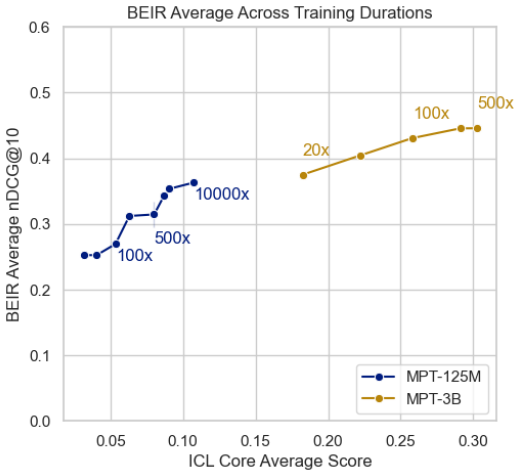

Figure S2: **Retrieval performance is highly correlated with In Context Learning (ICL) performance**. ICL Core Average Score uses the open-source MosaicML Evaluation Gauntlet.

Why don't any of our checkpoints achieve SOTA on BEIR or MTEB? In this study, we chose to focus on the scaling properties of pretrained models using a baseline finetuning approach. All of the datapoints represented by circles were finetuned on the same MS MARCO dataset with the same hyperparameters and a maximum sequence length of 128 tokens. This is a "lower bound" of performance, as most modern embedding models finetune on a maximum sequence length of > 512 tokens and use much more extensive finetuning datasets.

One example of a more extensive finetuning dataset is E5-Mistral-7B (Wang et al., 2023). The E5-Mistral-7B team finetuned Mistral-7B on a custom, closed source hard-negatives dataset that consists of 13 semi-open-source datasets (they mined their own hard negatives) including MS MARCO as well as synthetically generated GPT-3.5/4 data (Wang et al., 2023). We only trained our checkpoints on MS MARCO.

In this study, we don't derive phenomenological scaling laws; rather we report a strong trend. We save formal retrieval scaling laws for future work.

C Retrieval metrics

While there are many metrics used for retrieval, normalized discounted cumulative gain (nDCG) is a particularly common one. This metric indicates whether the retrieved documents are (1) relevant and whether (2) they are sorted in order of relevance to the query. Specifically, it is defined as:

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i+1)}$$

$$\mathrm{IDCG_p} = \sum_{i=1}^{|REL_p|} \frac{rel_i}{\log_2(i+1)}$$

[5] Llama 2 7B and 13B, Llama 3.2 1B and 3B, Gemma 2 3B, Llama 3.3 8B, Mistral 7B, Mistral 8x7B, OLMo 2 7B and 13B, OLMoE-1B-7B, DeepSeekMoE-3B-16B.

[6] With the minor exceptions of MosaicBERT (Portes et al., 2023), NomicBERT (Nussbaum et al., 2024), and ModernBERT (Warner et al., 2024).

where rel_i is graded relevance of the result at position i. Finally, the normalized discounted cumulative gain is defined as:

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCG_p} \tag{2}$$

D BEIR Retrieval Tasks

ArguAna Wachsmuth et al. (2018): consists of 1406 queries and 8674 documents in the test set. Data consists of single sentence "arguments" and single sentence "counterarguments" originally curated from the obscure online debate portal idebate.org. The task is to find documents that *refute* the claim in the query, which makes the task particularly difficult without instructions or instruction tuning.

ClimateFEVER Diggelmann et al. (2020): is similar in spirit to FEVER, ClimateFEVER is a dataset for verification of climatechange-related claims.

CQADupstackRetrieval Hoogeveen et al. (2015). The task is designed such that, given a single sentence title (the query), the model has to retrieve a duplicate document (title +body).

DBPedia Hasibi et al. (2017) is derived from Wikipedia pages.

FEVER Thorne et al. (2018) The original Fact Extraction and VERification dataset was collected semiautomaticallyas part of an automatic fact checking task. The task is to retrieve Wikipedia abstracts that support or refute a given claim.

FiQA2018 Maia et al. (2018): passages are were scraped from StackExchange posts under the Investment topic from 2009-2017. 648 queries and 57638 documents in the test set.

HotpotQA Yang et al. (2018) - multi-hop like questions derived from ? that require multiple separate wikipedia paragraphs to answer. 5447 queries.

MSMARCO Bajaj et al. (2016): Since all models were finetuned on MS MARCO, this is not considered "zero-shot."

NFCorpus Boteva et al. (2016): 323 queries and 3633 documents in the test set. Queries taken from Nutrition Facts website, annotated medical documents from PubMed are the documents.

NQ Kwiatkowski et al. (2019): natural questions is Google searches with answer spans from Wikipedia articles. 3,452 queries, search over 2,681,468 passages. While the original NQ dataset includes full articles, the MTEB BEIR version only uses wikipedia abstracts *and not full articles*.

QuoraRetrieval: 10,000 queries, and 522,931 queries as "documents" in the test set.

SCIDOCS Cohan et al. (2020) consists of scientific documents.

SciFact Wadden et al. (2020) contains 1.4K expert-written scientific claims. These are paired with paper abstracts.

Touche2020 Bondarenko et al. (2020): 49 queries, 382545 documents in the test set. conversational arguments. Use conclusion as title and premise as arguments. Note that only 49 queries is potentially quite noisy.

TREC-COVID Voorhees et al. (2021): CORD-19 dataset of published scientific articles dealing with the COVID-19 pandemic. 50 queries and 171,332 documents in the test set.

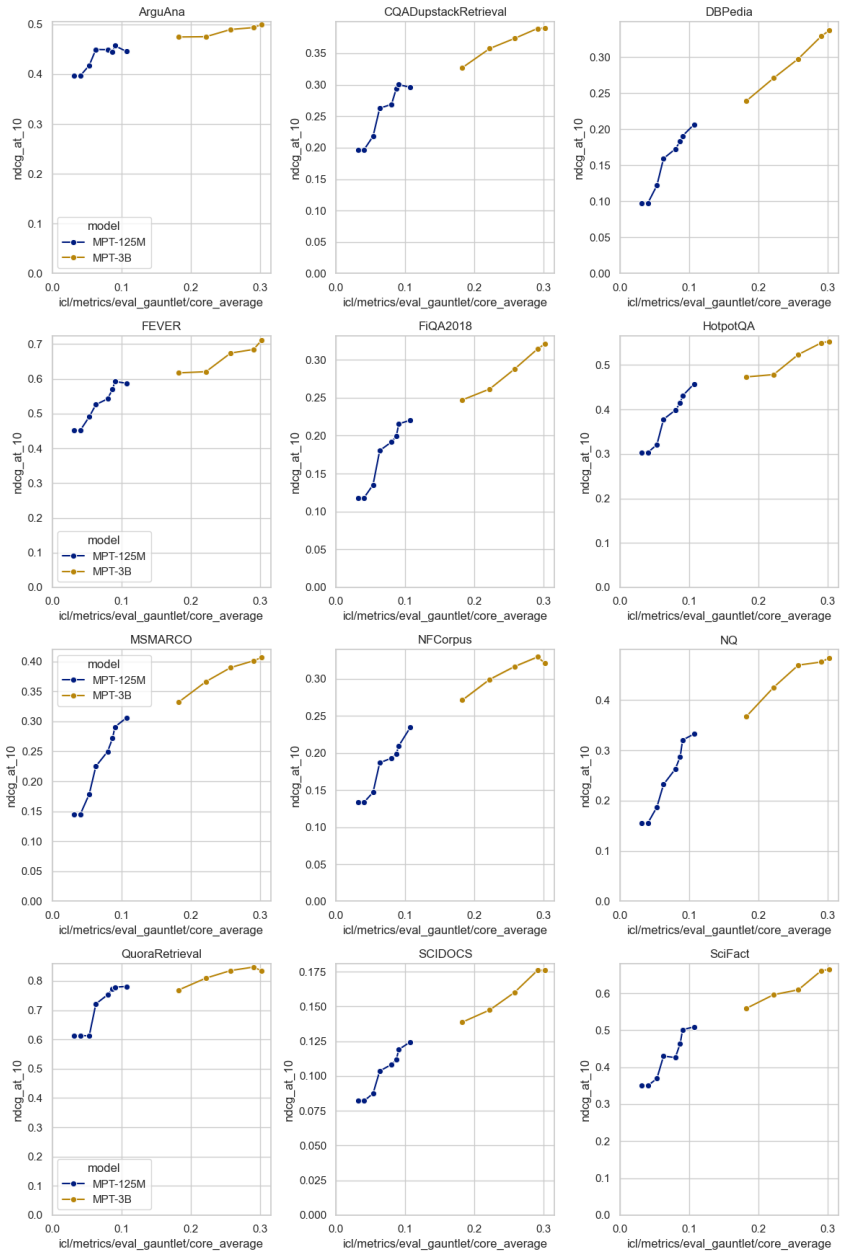

Figure S3: **Retrieval performance on individual BEIR tasks is highly correlated with ICL performance.** Same data as Figure S2.