GRANITE EMBEDDING MODELS

Granite Team *†
IBM Research AI

ABSTRACT

We introduce the Granite Embedding models, a family of encoder-based embedding models designed for retrieval tasks, spanning dense-retrieval and sparse-retrieval architectures, with both English and Multilingual capabilities. This report provides the technical details of training these highly effective 12 layer embedding models, along with their efficient 6 layer distilled counterparts. Extensive evaluations show that the models, developed with techniques like retrieval oriented pretraining, contrastive finetuning, knowledge distillation, and model merging significantly outperform publicly available models of similar sizes on both internal IBM retrieval and search tasks, and have equivalent performance on widely-used information retrieval benchmarks, while being trained on high-quality data suitable for enterprise use. We publicly release all our Granite Embedding models under the Apache 2.0 license, allowing both research and commercial use at `https://huggingface.co/collections/ibm-granite`.

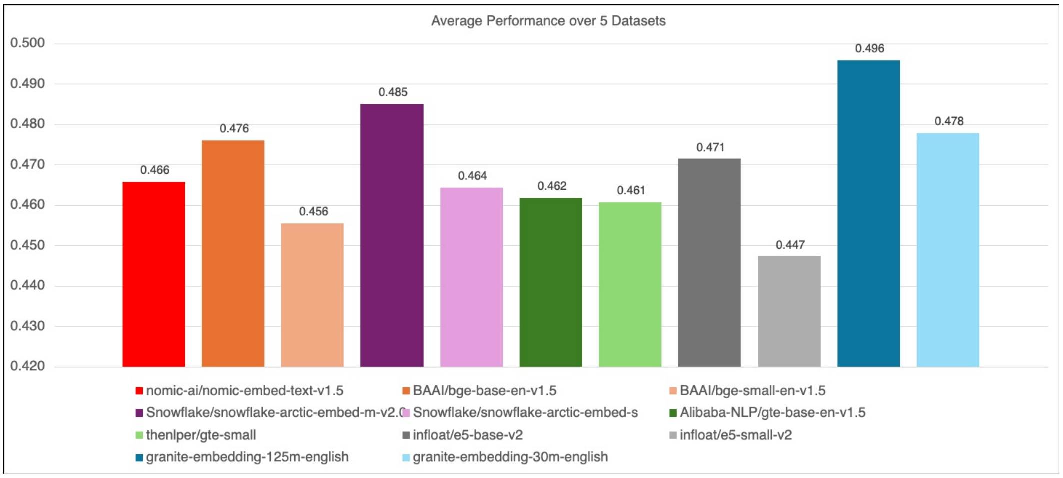

Figure 1: Average performance on the Granite embedding models (in blue) vs BGE, GTE, Snowflake, E5, and Nomic models on 5 QA and IR datasets: BEIR, ClapNQ, CoIR, RedHat, and UnifiedSearch (the last 2 are internal IBM datasets).

1 INTRODUCTION

The goal of text embedding models is to convert variable length text into a fixed vector, encoding the text semantics into a multidimensional vector in such a way that semantically close texts are close in the vector space, while dissimilar texts have a low similarity. These embeddings can then be used in a variety of tasks, most commonly in retrieval applications, where the relevance of a document to a given query can be determined by the similarity of their embeddings (Dunn et al., 2017; Xiong et al., 2020; Neelakantan et al., 2022)(Zamani et al., 2018; Zhao et al., 2020), but also in document clustering (Angelov, 2020) and text classification (Sun et al., 2019).

[*] See Contributions section for full author list.

[†] Please send correspondence to granite-inquiries@ibm.com.

While Large Language Models (LLMs) have been shown to be powerful sources for text embeddings (Lee et al., 2024; Wang et al., 2023), inference with these models are often impractical in an industrial setting, due to large latency and memory footprint. Encoder-based text embedding models (Wang et al., 2022; Xiao et al., 2023a; Chen et al., 2024; Merrick et al., 2024) are thus more popular, however they are often trained on data with non-commercial licenses, leading to restrictions in commercial deployment.

This report introduces Granite Embedding models, a family of encoder embedding models purpose-built for retrieval tasks. All Granite Embedding models have been trained on high quality, curated data, with data quality checks and screening to remove personal information and profane language. We release these models under the Apache 2.0 license, permissible for both commercial and research applications. We release 5 models of varying language support, sizes, and vector types[1]:

- Dense English Models: 30M and 125M parameter models, specialized for English-only retrieval applications
- Dense Multilingual Models: 107M and 278M parameter models, finetuned for multilingual retrieval for 12 languages
- Sparse English Model: a 30M model adapted for sparse retrieval capabilities in English

The granite-embedding models are specifically adapted for enterprise usage, offering strong performance for a variety of inference budgets. The base-size models (12 layers) have strong performance on both academic and enterprise tasks, and the light-weight 6-layer models provide good performance with very low latency and memory requirements.

The structure of the paper is as follows: Section 2 describes the model architecture and training background, the training data and its curation is described in Section 3. Section 4 describes in detail the training recipes for the Granite Dense English Models, while Section 5 describes the dense multilingual models and Section 6 describes the sparse English model. Finally, Section 7 presents a comprehensive evaluations of the granite embedding models, comparing their performance with other open source encoder embedding models.

2 Granite Embedding Models

The Granite Embedding collection delivers innovative sentence-transformer models purpose-built for retrieval-based applications. Featuring a bi-encoder architecture (Reimers & Gurevych, 2019), these models generate high-quality, low-dimensional embeddings for textual inputs – such as queries, passages, and documents – enabling seamless comparison through cosine similarity. In this Section, we describe the architectural details of our models, along with a general overview of our training methodology.

Model Architecture The Granite Embedding family consists of five encoder-only models of varying sizes and vocabularies for a diverse set of use-cases, featuring English and Multilingual models targeted for retrieval applications. The English models are RoBERTa-like (Liu et al., 2019) models while the Multilingual models follow an XLM-RoBERTa (Conneau et al., 2020) configuration, all trained internally at IBM Research. The Multilingual models are targeted towards 12 languages, however, they can be finetuned on other languages covered the XLM-RoBERTa vocabulary (approximatively 100 languages).

We release five high performing base-sized models, *granite-embedding-125m-english* and *granite-embedding-278m-multilingual*, as well as smaller, faster counterparts, *granite-embedding-30m-english*, *granite-embedding-107m-multilingual* and *granite-embedding-30m-sparse* suitable for diverse resource and latency budgets.

Dense Granite embeddings models use the final hidden state of the CLS token, or $<s>$ for Roberta-based models, as the embedding representation (also known as CLS pooling), which we find to

[1] The main difference between the sizes of the models comes from the vocabulary: the multilingual models have a large vocabulary of 250K tokens, while the English models have 50k token vocabulary (Table 2); the networks themselves have the sane number of parameters. We have reused the RoBERTa (Liu et al., 2019) and XLM-RoBERTa (Conneau et al., 2020) vocabularies in these models.

perform better than mean-pooling strategy commonly used in other works (Wang et al., 2022; Li et al., 2023; Nussbaum et al., 2024). The sparse Granite embedding model uses max-pooling across all sequence tokens to produce a variable length weighted bag-of-words like output. The maximum context length of all Granite embedding models is 512 tokens, and longer sequences are truncated to this length.

Detailed specifications of the architecture of each model is shown in Table 2.

Model	30M	125M	107M	278M
Embedding size	384	768	384	768
Layers	6	12	6	12
Intermediate size	1536	3072	1536	3072
Max. Sequence Length	512	512	512	512
Vocabulary Size	50265		250002	
Target Languages	En		En, Ar, Cs, De, Es, Fr, It, Ja, Ko, Nl, Pt, Zh	

Table 1: Architectural Details for Granite Embedding Models

Training Recipe Embedding models are trained using a contrastive learning objective (Gao et al., 2021), which brings the embeddings of a query closer to those of relevant passages, and pushes them further away from non-relevant ones. In this work, we use an improved contrastive loss proposed in Li et al. (2023) which introduces an additional bidirectional signal to expand negatives. Specifically, for a batch of triples $([q_i, (p_{ij})_j])_i$ consisting of a query and a set of passages – without loss of generality, we can assume that p_{i0} is a positive passage for query i, while $p_{ij}, j > 0$ are negative passages – we define the contrastive loss as:

$$\mathcal{L}_C = -\frac{1}{n}\sum_{i=1}^{n}\log\frac{e^{s(q_i,p_{i0})}}{Z_i} \tag{1}$$

$$Z_i = e^{s(q_i,p_{i0})} + \alpha\sum_{j>0} e^{s(q_i,p_{ij})} + \beta\sum_{i'\neq i} e^{s(q_i,q_{i'})} + \gamma\sum_{j>0} e^{s(p_{i0},p_{ij})} \tag{2}$$

where $s(q, p)$ is a the temperature-scaled cosine similarity between the embeddings of q and p:

$$s(q,p) = \frac{1}{\tau}\frac{\mathbf{E}(q)\cdot\mathbf{E}(p)}{\|\mathbf{E}(q)\|\|\mathbf{E}(p)\|} \tag{3}$$

where $\mathbf{E}(p)$ is the embedding of text p[2], here α, β, and γ are added so that we can remove any of these components easily (they can be either 0 or 1). Contrastive training is conducted using paired data sourced or generated from a diverse set of domains. Similar to prior work (Wang et al., 2022; Merrick et al., 2024; Xiao et al., 2023a; Nussbaum et al., 2024) we use both weakly-paired data from web sources, along with high quality annotated data and targeted synthetic data. We use in-batch negatives to better approximate the contrastive signal, and also include hard negatives for a subset of our data consisting of high-quality annotated pairs, mining them whenever needed.

We explore additional techniques to improve the performance of our models. For our English models, we find it helpful to include an initial **retrieval-oriented pretraining** step, similar to Xiao et al. (2022), which uses an asymmetric model structure and masked auto-encoding to improve embedding quality. We also find that **knowledge distillation** techniques, particularly distilling the distribution of similarity scores from larger models, are extremely effective in improving the performance of

[2] Note that, for high temperatures τ, this loss is a smooth version of maximizing

$$s(q_i, p_{i0}) - \max\left(\alpha\max_{j>0} s(q_i,p_{ij}), \beta\max_{i'\neq i} s(q_i,q_{i'}), \gamma\max_{j>0} s(p_{i0},p_{ij})\right)$$

our models: *granite-embedding-125m-englis* is distilled from a larger decoder-based embedding model, *granite-embedding-278m-multilingual* uses self-distillation to improve performance and all our small models are distilled from the larger encoder-based embedding models. Finally, we use **model merging** Xiao et al. (2023b) to help adapt our models to specific domains without losing general performance. More details on training are provided in the subsequent sections.

3 Training Data

Granite Embedding Models are trained on two types of data:

1. weakly paired data mined from the web with in-batch negatives and
2. high quality, annotated data with hard negatives for finetuning derived from three key sources:
    (a) publicly available paired data,
    (b) IBM-internal paired data targetting specific technical domains, and
    (c) IBM-generated synthetic data.

For governance, our data undergoes a data clearance process subject to technical, business, and governance review. This comprehensive process captures critical information about the data, including, but not limited to, its content description, intended use, data classification, licensing information, usage restrictions, as well as an assessment of sensitive information (i.e, personal information).

Weakly Paired Data with In-Batch Negatives The bulk of this data consists of title-body pairs obtained from diverse web sources such as Wikipedia, StackExchange, Semantic Scholar, Arxiv and PubMed. We also include citation pairs (SPECTER (Cohan et al., 2020), Semantic Scholar (Lo et al., 2020)), duplicate questions (StackExchange), and question answer pairs (PAQ (Lewis et al., 2021), WikiAnswers (Fader et al., 2014), SearchQA (Dunn et al., 2017)). For multilingual models, a subset of the English data is used, along with title-body pairs from mC4, multilingual Wikipedia, and multilingual Webhose, and Machine Translations of StackExchange. While training with these larger datasets, we only include in-batch negatives to provide a contrastive signal.

Annotated Data with Hard Negatives We use high-quality open source question-answer pairs from NQ (Kwiatkowski et al., 2019), SQuAD (Rajpurkar et al., 2018) and HotpotQA (Yang et al., 2018), and fact verification pairs from FEVER (Thorne et al., 2018). Notably, while other open-source models trained on the popular MS-MARCO retrieval dataset (Bajaj et al., 2018), we chose not to train on it given it is designed for research purposes only rather than commercial use. For our multilingual models, we also include data from MIRACL (Zhang et al., 2023), TyDiQA (Clark et al., 2020) and Sadeem Question Answering (SadeemAI, 2024). Additionally, we find it helpful to include cross-lingual, translated parallel corpora to help improve multilingual performance. For enterprise applications, we create title-body pairs from IBM internal datasets targeting specific technical domains. For all these datasets, we mine additional hard negatives for a stronger contrastive signal (except MIRACL and TyDi, which have annotated hard negatives).

Synthetic Data In addition to the above sources, we generate high quality synthetic multilingual queries with LLMs[3] (Jiang et al., 2024; Abdin et al., 2024) for Wikipedia paragraphs in all our target languages. We carefully design prompts that generate a diverse set of queries for a given input paragraph, such as simple keyword search queries, queries that can be partially/completely answerable from the given paragraph, facts that are validated by the paragraphs, summaries of the paragraph, etc. in order to help the model learn different notions of similarity. We also generate hard negatives for these queries, where we instruct the model to modify the positive passage such that it maintains the context of it, but does not answer the given query.

Hard Negative Mining For high quality datasets that are not annotated with hard negatives, we mine them using a pre-existing lightweight embedding model trained at IBM Research. We calculate the similarity of non-relevant passages from the same dataset with the given query, and sample

[3] `mistralai/Mixtral-8x22B-Instruct-v0.1,microsoft/Phi-3.5-MoE-instruct`

negatives from the top-100 ranked passages. We filter out false negatives by discarding passages whose similarity to the positive passage lies above a threshold. For fine tuning, we use a single hard negative, along with random in-batch negatives for the annotated data, and find that increasing the number of hard negatives does not significantly help performance. For distillation, we use up to three hard negatives and find out it improves the performance over single hard negative. We suspect that this is because more hard negatives fill the gap between hard negative similarity and in-batch negative similarity. Therefore provides more signal for distillation.

Data Sampling During training, each batch is sourced from a single dataset. To address the imbalance between the sizes of the datasets used for contrastive training, we sample batches from each dataset D_i proportionately to its size, $|D_i|$, where the sampling probability is defined by:

$$p_i = \frac{|D_i|^\alpha}{\sum_i |D_i|^\alpha} \tag{4}$$

where α is a hyperparameter. We observe α value of 0.5 and 0.9 to result in best scores for English and multilingual models resp.

4 Dense Retrievers: English

4.1 granite-embedding-125m-english

granite-embedding-125m-english is a dense bi-encoder embedding model that has the same architecture as RoBERTa-base (Liu et al., 2019). It performs well on many enterprise use cases while maintaining competitive scores on academic benchmarks such as BEIR. The starting point of fine tuning this model is a RoBERTa-base like model, called WatBERT, internally trained by IBM using carefully curated data. The model is then further fine tuned through two steps: (1) retrieval oriented pretraining, (2) knowledge distillation from decoder based embedding model.

Retrieval Oriented Pretraining We first apply retrieval oriented pretraining to the WatBERT model similar to Retro-MAE (Xiao et al., 2022). The training follows a masked auto-encoding workflow. The input sentence is masked twice with different ratios. One moderately masked input (15%-30%) is used by the encoder model where the sentence embedding is generated from the [CLS] token. The sentence embedding joins with the other aggressively masked input (50%-70%) and is used by a single-layer transformer to recover the original sentence via masked language modeling. We train the model with 56 million sentences from Wikipedia, BookCorpus and StackExchange.

Decoder based teacher model We further fine tune the Mistral-7B-Instruct-v0.2 (Jiang et al., 2023) using contrastive training as our teacher model. Similar to (Wang et al., 2024a), for each query-document pair (q, p), we apply an instruction template to the original query to generate a new one:

$$q_{inst} = Instruct: \{task\_definition\} Query: q \tag{5}$$

where "{task_definition}" is a placeholder for a one-sentence description of the embedding task. We then append an [EOS] token to the end of the query and document and feed them into the Mistral model to obtain the query and document embeddings by taking the [EOS] vector of the last layer.

We train two separate embedding models from the Mistral-7B-Instruct-v0.2 and create the final teacher model by merging the two models together. The first model is trained using three million pairs of weakly paired data with in-batch negatives. The second model is trained using one million annotated data with one hard negative.

Distill from decoder based model Using the same methodology as described in Section 4.2, we then distill the knowledge of mistral-7b-instruct-v0.2 embedding model into *granite-embedding-125m-english*. Notably, the similarity score of the decoder model is calculated from the [EOS] token while the similarity score of the encoder model is calculated from the [CLS] token. The distillation also happens in two stages. In the first stage, weakly paired data with in-batch negative is used while in the second stage, high quality annotated data with three hard negatives is used.

4.2 GRANITE-EMBEDDING-30M-ENGLISH

granite-embedding-30m-english is an efficient embedding model distilled from an independently trained encoder model similar to *granite-embedding-125m-english*. This model is about 4 times faster than *granite-embedding-125m-english*, while maintaining good performance, and is highly suitable to use in latency-constrained applications. This model is trained in three steps: (1) Retrieval Oriented Pretraining, 2) contrastive based distillation on a large corpus of data (3) targeted enterprise-domain adaption.

Retrieval Oriented Pretraining We prime our model with RetroMAE Based Distillation. First, we incorporate supervision from the RetroMAE-pretrained encoder described in 4.1 using distillation. We substitute the standard masked language modeling loss for the encoder with a distillation loss between the student's and teacher's output distribution, while keeping the unmodified decoder loss from Xiao et al. (2022). We use Wikipedia, BookCorpus and StackExchange in this step.

Contrastive Based Distillation We tune the embeddings by distilling the distribution of similarity scores of the teacher model. Specifically, using the same notations from Section 2, the training objective minimizes the cross entropy between the teacher's distribution P_t of similarity scores between pairs and the student's distribution, P_s. Following Hinton et al. (2014), we also scaled the score distribution of both teacher and student by a temperature, τ:

$$\mathcal{L}_{KD} = -\sum_{i=1}^{n}\sum_{j=1}^{m} P_t(q_i, p_{i0}) \log P_s(q_i, p_{i0}) \tag{6}$$

$$P_s(q_i, p_{i0}) = \frac{e^{s_s(q_i, p_{i0})/\tau}}{Z_{s,i}} \tag{7}$$

$$p_t(q_i, p_{i0}) = \frac{e^{s_t(q_i, p_{i0})/\tau}}{Z_{t,i}} \tag{8}$$

Here, $s_s(q_i, p_{i0})$ and $s_t(q_i, p_{i0})$ represent the similarity scores between questions and reference passages $\{q_i, p_{i0}\}_i$ defined in Equation 3, for the student and teacher respectively, and $Z_{s,i}$ and $Z_{t,i}$ are computed as in Equation 2, with $\beta = 0$ and $\gamma = 0$.

Utilizing the similarity scores for distillation instead of the distilling the text embedding vectors directly allows for effective knowledge transfer between models of different embedding sizes, without introducing additional parameters for embedding projection. For datasets without a mined or annotated hard negative, we find that perturbing the positive passage slightly (through deletion or swapping of spans of text) is an effective and inexpensive way to create a richer distribution of similarity scores, leading to better performance after distillation. While the resulting passage may not necessarily be a *true negative*, the lack of an explicit contrastive signal differentiating between positive and negative passages during score distillation allows this to be an effective way of using a large quantity of unsupervised data.

Targeted Enterprise-Domain Adaption In order to strengthen performance on specific enterprise domains, we conduct another round of distillation on IBM-internal paired data. We use model merging (Xiao et al., 2023b) between this domain-adapted model and the model trained on all the data, resulting in the best overall performance in both enterprise-specific and general-purpose domains.

5 DENSE RETRIEVERS: MULTILINGUAL

We develop multilingual variants of Granite Embedding models, targeted towards 12 languages, trained on a large amount of multilingual data including synthetically generated pairs. These models are trained by gradually expanding the number of languages in the training data: we start with older, un-released versions of these models trained on only 6 languages (English, French, German, Japanese, Portuguese, and Spanish), and further train these models on annotated or synthetic data from all our target languages (aforementioned languages along with Arabic, Czech, Dutch, Italian, Korean and Chinese). We find that this helps maintain the performance of the original models on the languages they are trained on, while improving performance on new languages added in the mix. For the multilingual models, a retrieval-oriented pretraining step does not improve the model's performance after finetuning, so we omit this initial step.

5.1 GRANITE-EMBEDDING-278M-MULTILINGUAL

granite-embedding-278m-multilingual is an effective, base-size model specifically trained for retrieval applications in multilingual text settings for 12 Languages. To build this model, we start from an in house XLM-RoBERTa like model, trained on 100 languages, and first train it for retrieval on a 6-languages using a stage-wise contrastive training approach, common in previous works. This is followed by extending finetuning on all our target langauges. Finally, we conduct a self-distillation step, which we find to be extremely useful in improving the performance of our model without additional data or an external model.

Stage-wise Contrastive Training First, we train a six-language model using a stage-wise training approach, as done in prior work (Xiao et al., 2023a; Chen et al., 2024; Wang et al., 2022; Li et al., 2023; Zhang et al., 2024). Specifically, we start with an *unsupervised pretraining* phase, where we train on a large corpus of naturally occuring pairs (eg. title-body) mined from web sources. Here we use a large batch size and in-batch negatives for effective learning. We then conduct a *supervised finetuning* step using annotated pairs with a single hard negative. This results in a model with good performance on English, French, German, Japanese, Portuguese, and Spanish.

Target Language Finetuning To expand the scope of this model, we conduct further finetuning on all our 12 target languages. This is done with annotated and synthetic paired data with a single hard negative. We find this strategy to be more effective at maintaining prior performance than training a 12-language model from scratch.

Self Distillation We find that an additional training step further improves the performance of this model. Self Knowledge Distillation is a technique where the knowledge of a model is transferred into another model of the same size using its temperature-scaled output distribution. Prior work has shown this technique to be an effective method of progressive performance improvement (Furlanello et al., 2018; Kim et al., 2021), and we interpret this technique as a way to reduce the effect of any noisy labels. Specifically, we distil the output score distribution of the finetuned 12-language model into a student initialized from the same teacher on the distillation loss of Equation 6. We use all the finetuning data available for this distillation step, including our synthetic data and enterprise-domain data.

5.2 GRANITE-EMBEDDING-107M-MULTILINGUAL

granite-embedding-107m-multilingual is an efficient multilingual model, almost 4 times faster than *granite-embedding-278m-multilingual* while maintaining good performance. This model is distilled from components of the *granite-embedding-278m-multilingual* model: first, we distill the intermediate 6-language base model using data in these 6 languages, then continue distilling from the 12-layer model on all languages.

Like *granite-embedding-30m-english*, this model is trained by distilling with all the available data, using a single hard negative when available, and perturbing the positive to get a richer probability distribution for datasets without hard negatives. We also use in-batch negatives to better approximate the contrastive signal.

Similar to it's larger counterpart, *granite-embedding-107m-multilingual* is constructed by first training a 6-language model, by distilling from the intermediate base model on all the training data for English, French, German, Japanese, Portuguese, and Spanish. Then, we distill from *granite-embedding-107m-multilingual* on all the 12 target languages. Distilling in stages yields a better overall performance than distilling all 12 languages in a single step.

6 Sparse Retrievers: English

The Granite dense retriever models, described in 4 , produce a fixed-size embedding vector for a query or document. Alternatively, Granite sparse retriever model is trained to produce a variable length bag-of-word like dictionary, containing expansions of sentence tokens and their corresponding weights. This sparse representation could be useful for exact matching of terms and could be efficiently ingested as an inverted index as shown by Zamani et al. (2018); Bai et al. (2020); Dai & Callan (2019); Formal et al. (2021b).

6.1 Granite-Embedding-30m-Sparse

granite-embedding-30m-sparse is a 6 layer model, similar in architecture to *granite-embedding-30m-english*, and is trained following the first two training steps of the dense counterpart 4.2:(1) Retrieval Oriented Pretraining (2) contrastive-based distillation on a large corpus of data.

Sparse models are popularly trained to predict the importance of each vocabulary term based on the logits of masked-language model head (Bai et al., 2020; Formal et al., 2021b). We start with a 6 layer WatBERT model, apply Retro-MAE based distillation, and then further train the language model head using contrastive-based knowledge distillation from a larger (125m) dense retriever teacher. We experiment with Margin-MSE loss (Hofstätter et al., 2021), but find Cross Entropy to be more effective in our setup, possibly because our teacher is a dense retriever. Further, we introduce another loss, as described below, to train our sparse model.

Term Weight Pooling Strategy We follow Formal et al. (2021a) and use max-pooling to compute the importance w_j for each vocabulary token j for input sequence I:

$$w_j = \max_{i \in I} \log(1 + \text{ReLU}(w_{ij})) \tag{9}$$

Learning Sparse Representations We observe that minimizing the FLOPS loss Paria et al. (2020), as shown in SPLADE (Formal et al., 2021b;a), did not lead to a sufficiently sparse model. We hypothesize this being an artifact of our starting point, the 6 layer WatBERT model, as the 6 layer WatBERT model is distilled from the 12 layer WatBERT 4.1 using multi-head self-attention relation distillation (Wang et al., 2021) which does not train the Masked Language Modeling head. So, in addition to the FLOPS loss we introduce a NORM loss that learns to minimize the total norm of the sentence with the intuition being this would further bring the term weight close to zero, as follows:

$$\mathcal{L}_{NORM} = \sum_j \max_{i \in I} \log(1 + \text{ReLU}(w_{ij})) \tag{10}$$

Overall Loss We jointly optimize the sparse model with contrastive distillation 6, FLOPS regularization, and NORM regularization 10 losses:

$$\mathcal{L} = \mathcal{L}_{KD} + \lambda_q \mathcal{L}^q_{FLOPS} + \lambda_p \mathcal{L}^p_{FLOPS} + \sigma_q \mathcal{L}^q_{NORM} + \sigma_p \mathcal{L}^p_{NORM} \tag{11}$$

where λ_q and λ_p are FLOPS regularization weights and σ_q and σ_p are total norm regularization weights for query and passage resp.

7 Evaluation

We evaluate the performance of the Granite Embedding models on a variety of retrieval tasks, spanning multiple domains and languages. Figure 1 shows comparative performance of the english Granite Embedding models with other open source models. Results show the Granite Embedding models achieve higher scores in average than the all other models. In this section we show results that further show that the Granite Embedding models are faster, achieve higher average accuracy

Model	Argua-Ana	Climate Fever	CQA-Dup-stack	DB-Pedia	Fever	FiQA	HotPot-QA	MS-Marco	NF-Corpus	NQ	Quora	Sci-Fact	Sci-docs	Touche	Trec-Covid	**Avg.**	**Avg. (no MS-Marco)**
e5-small-v2	41.7	22.9	37.1	41.3	81.6	37.4	66.6	41.5	32.5	59.1	85.7	68.9	17.8	27.1	74.5	49.0	49.6
bge-small-en-v1.5	59.6	31.8	39.1	40.0	86.6	40.3	69.9	40.8	34.3	50.2	88.8	71.3	20.5	26.0	75.9	51.7	52.5
e5-base-v2	44.5	26.6	38.5	42.2	85.0	39.9	69.2	41.8	35.4	58.2	86.6	71.9	18.7	26.4	69.6	50.3	50.9
bge-base-en-v1.5	63.6	31.2	42.4	40.8	86.3	40.7	72.6	41.4	37.4	54.2	88.9	74.0	21.7	25.7	78.1	53.3	54.1
granite-embedding-30m-english	56.4	30.3	44.3	36.0	85.5	36.9	62.9	30.7	33.7	51.6	86.7	71.3	22.5	24.0	63.1	49.1	50.4
granite-embedding-125m-english	58.4	33.2	48.2	39.4	88.2	44.9	67.8	32.5	37.3	58.0	87.8	74.7	24.2	20.2	69.3	52.3	53.7
multilingual-e5-small	39.1	22.6	36.1	37.8	75.3	33.3	65.1	41.0	31.0	56.3	86.9	67.7	13.9	21.2	72.6	46.6	47.0
multilingual-e5-base	44.2	23.9	38.5	40.4	79.4	38.2	68.6	42.3	32.5	60.0	87.7	69.3	17.2	21.4	69.8	48.9	49.4
granite-embedding-107m-multilingual	51.7	25	39.9	32.8	83.2	28.4	58.8	28.3	27	50.9	85.8	63.3	17.4	24.9	62.8	45.3	46.6
granite-embedding-278m-multilingual	55.2	29.5	42.7	34.6	84.9	35.7	61.5	29.9	28.9	53.4	86.9	17.7	65.9	27.3	68.6	48.2	49.5
splade-v3-distilbert	48.4	22.8	-	42.6	79.6	33.9	67.8	-	34.8	54.9	81.7	14.8	68.5	30.1	70	50.0†	50.0*
granite-embedding-30m-sparse	54.7	25.5	42.1	37.2	82.0	35.2	68.1	30.3	32.3	50.8	85.1	71.4	21.3	22.8	71.9	48.7	50.6*

Table 2: BEIR benchmark (Thakur et al., 2021) nDCG@10 scores. † Average is not directly comparable as splade-v3-distilbert (Lassance et al., 2024) does not report BEIR scores on CQADupStack and MS-MARCO. * Average does not include CQADupstack.

over a variety of benchmarks, and that they not only maintain competitive scores on benchmarks such as MTEB, but also out-perform the other models on Code Retrieval.

We compare our encoders with other state-of-the-art embedding models of similar parameter size. Specifically, we compare *granite-embedding-125m-english* to the BGE Base (Xiao et al., 2023a) and E5 Base Models (Wang et al., 2022), which have identical architecture as the granite base models, with a smaller vocabulary size. *granite-embedding-30m-english* is compared to the small versions of the BGE and E5 model family: note that these models have double the number of layers than the Granite 30M models. We compare the multilingual models against the Multilingual E5 Wang et al. (2024b) Base and Small models respectively. For sparse embeddings, we compare *granite-embedding-30m-sparse* with SPLADE-v3-DistilBERT Lassance et al. (2024), a 6-layer model with larger 768 embedding size.

7.1 English Retrieval Performance

All Granite Embedding models are evaluated on retrieval tasks from the MTEB benchmark (Muennighoff et al., 2022), which consists of datasets and evaluation from BEIR (Thakur et al., 2021). This consist of 15 datasets spanning multiple domains, and is used to test the model's ability to find the relevant document for a given query. The experiment results are reported in Table 2. We report the nDCG@10 scores on each dataset, showing strong performance relative to other open-source models of similar sizes. We also note the average performance on all tasks except MS-MARCO retrieval Bajaj et al. (2018), for a fair zero-shot comparison, as other embedding models train on MS-MARCO, but our models do not, due to unfavorable license. Despite being trained on less data, and only permissibly licensed public datasets, our models show strong performance.

While our models are purpose-built for retrieval applications, we also evaluate them on the entire MTEB benchmark (including performance on non-retrieval tasks such as classification, clustering, pair clustering, re-ranking, retrieval, sentence similarity and summarization) in Appendix B to measure the quality of embeddings for non-retrieval tasks.

7.2 Multilingual Retrieval Performance

The Multilingual Granite Embedding models are evaluated on the Miracl Benchmark (Zhang et al., 2023) and the Mintaka Retrieval task (Sen et al., 2022), both of which measure the performance of monolingual retrieval on multiple languages. Miracl is a high-quality text retrieval benchmark sourced from Wikipedia, in 18 langauges. Mintaka Retrieval comprises of Wikipedia-based questions translated into 8 langauges. We report the nDCG@10 scores for these models on both of these tasks in Table 3 and Table 4.

Model	ar	bn	de	en	es	fa	fi	fr	hi	id	ja	ko	ru	sw	te	th	yo	zh	**Avg.**
multilingual-e5-small	71.4	68.3	48.8	48	50.8	53.3	73.3	47.1	55.1	50.7	63.6	61.2	59	68.5	81.3	74.9	45.3	45.9	59.3
multilingual-e5-base	71.6	70.2	52.1	51.2	51.5	57.5	74.3	49.7	58.3	51	64.7	62.3	61.6	71.1	75.1	75.3	70.6	57.5	62.5
granite-embedding-107m-multilingual	62.5	65.5	45.1	46.7	46.6	50	65.6	47.6	42.1	45.5	59.2	58.4	48.3	59.1	78.2	71.8	64.7	48.4	55.9
granite-embedding-278m-multilingual	64.2	68.1	48.1	49.4	49.7	50.2	67.5	49.9	46.1	47.2	62.8	59.2	52.3	61.3	79.2	73.3	68.7	52.6	58.3

Table 3: MIRACL benchmark (Zhang et al., 2023) nDCG@10 scores.

Model	ar	de	es	fr	hi	it	ja	pt	**Avg.**
multilingual-e5-small	21.2	25.6	26.4	25	21.1	26.2	20.7	24.4	23.8
multilingual-e5-base	23.1	29.8	29.9	30.9	22.7	29.8	22.9	30.6	27.5
granite-embedding-107m-multilingual	18	24.9	25.4	24.5	17.3	24.4	20.4	25.7	22.6
granite-embedding-278m-multilingual	18.3	25.7	26	25.3	18.3	24.7	21.2	26.4	23.2

Table 4: Mintaka Retrieval (Sen et al., 2022) nDCG@10 scores.

We also evaluate the multilingual embedding models on multilingual retrieval tasks from the MTEB library. For brevity, we only include the per-language average scores in Table 5, and include detailed results on all tasks in Appendix C. As shown, the Granite Models show comparable performance with respect to other open source models, except on Chinese. This is because we do not include the in-domain training data from DuReader (Qiu et al., 2022) or T2Retrieval (Xie et al., 2023), which are used by other multilingual models during their training.

Model	En	De	Es	Fr	Ja	Ar	Ko	Zh
multilingual-e5-small	46.6	75.7	51.0	50.3	62.3	68.6	75.1	59.9
multilingual-e5-base	48.9	76.6	52.3	55.6	64.5	69.7	76.4	61.6
granite-embedding-107m-multilingual	45.3	70.3	48.7	51.1	59.0	63.2	70.5	40.8
granite-embedding-278m-multilingual	48.2	71.2	52.6	54.1	61.7	64.2	71.8	45.2

Table 5: Multilingual MTEB Retrieval (Muennighoff et al., 2022) nDCG@10 scores. Average scores per language shown.

7.3 Code Retrieval Performance

We evaluate the model's code retrieval capability on the COIR benchmark Li et al. (2024), consisting of 10 datasets across 7 domains in Table 6. Granite Embedding models out-perform models of the same size, despite the fact that unlike other models, we do not include any code retrieval data in the training of our models, leading to purely zero-shot evaluation for our models.

7.4 Performance on IBM-related Benchmarks

The Granite Embedding models were developed primarily with customer applications in mind. We thus evaluate the performance of the English Granite Embedding models on three IBM tasks: ClapNQ Retrieval (Rosenthal et al., 2025), Red Hat Query Retrieval and Unified Search.

ClapNQ Retrieval consists of queries and passages, humanly modified from the Natural Questions Kwiatkowski et al. (2019) dataset, selected such that the answers are long and non-contiguous; we evaluate performance on the 301 answerable queries from the test set, as described in Rosenthal et al. (2025).

Red Hat Query Retrieval is an internal dataset that tests a models's ability to identify existing solutions to customer tickets regarding Red Hat products. This is formulated as a re-ranking task, where the model is used as a re-ranker over the top-100 passages selected with BM25. This test comprises of 466 test queries and a search corpus of 666K passages.

Model	APPS	CosQA	SynText-2SQL	CodeSearchNet	CodeSearchNet CCR	CodeTrans-Contest	CodeTrans-DL	Stack Overflow QA	FeedBack ST	FeedBack MT	**Avg.**
e5-small-v2	4.4	29.7	51.9	55.5	51.3	53.3	30.8	83.5	71.2	39.8	47.1
bge-small-en-v1.5	5.6	32.1	45.3	72.6	47.9	48.2	25.7	78.1	67.8	35.1	45.8
e5-base-v2	11.5	32.6	51.4	62.7	56.9	62.5	21.9	87.9	74.5	41.6	50.3
bge-base-en-v1.5	6.5	33.7	45.3	72.0	50.9	45.7	23.5	80.2	70.0	33.7	46.1
granite-embedding-30m-english	6.2	35.5	44.7	58.7	49.1	57.5	26.9	83.9	69.4	37.8	47.0
granite-embedding-125m-english	11.8	36.6	48.4	55.1	47.6	66.7	29.6	89.9	75.3	42.1	50.3
multilingual-e5-small	12.0	29.6	46.2	52.5	55.7	61.1	32.3	81.9	72.7	41.3	48.5
multilingual-e5-base	20.9	31.1	52.3	46.6	57.0	43.1	29.9	85.1	72.6	43.2	48.2
granite-embedding-107m-multilingual	4.5	28.3	39.3	46.3	38.3	43.7	18.2	66.6	58.5	23.5	36.7
granite-embedding-278m-multilingual	6.1	33.5	47.3	52.9	43	60.8	32.1	78.4	67.7	31.4	45.3

Table 6: COIR benchmark (Li et al., 2024) nDCG@10 scores.

Model	Red Hat (R@10)	ClapNQ (nDCG@10)	Unified Search (MRR@5)	**Avg.**
e5-small-v2	40.0	60.3	27.2	42.5
bge-small-en-v1.5	41.2	59.4	28.3	43.0
gte-small	42.2	60.3	33.0	45.2
snowflake-arctic-embed-s	42.1	62.3	35.4	46.6
e5-base-v2	41.4	62.5	31.2	45.0
bge-base-en-v1.5	41.2	65.3	28.0	44.8
gte-base-en-v1.5	40.2	65.1	29.1	44.8
nomic-embed-text-v1.5	40.9	68.5	28.2	45.9
snowflake-arctic-embed-m-v2.0	43.9	61.7	29.2	44.9
granite-embedding-30m-english	44.0	64.8	34.0	47.6
granite-embedding-125m-english	44.0	71.4	30.0	48.4

Table 7: IBM Benchmark evaluation. Reports nDCG@10 score for ClapNQ (Rosenthal et al., 2025), recall@10 score for Redhat Search and MRR@5 score for Unified Search

Unified Search test is another IBM internal benchmark based on IBM product documentation, used to evaluate the model's ability to retrieve the most relevant support document for a given user query regarding IBM software/hardware products. The test contains 402 queries, with a corpus of 1.7M documents.

We compare the performance of our models to other open English retrievers like Nomic Embeddings (Nussbaum et al., 2024), BGE models (Xiao et al., 2023a), E5 models Wang et al. (2022),Arctic Embeddings (Merrick et al., 2024) and GTE models (Li et al., 2023). As shown in Table 7, the Granite Embedding models outperform all other models in these internal tasks.

7.5 Retrieval Time

The latency of retrieval is an important aspect of the large-scale applications of embedding models. As shown in Table 8 The Granite Embedding models, specifically the smaller variants, have lower retrieval time per query compared to other open source models with the same vector size, while maintaining good performance. The result is based on evaluating an IBM internal benchmark with single A100 GPU.

Model	Vector Size	Time Per Query (s)
e5-small-v2	384	0.31
bge-small-en-v1.5	384	0.31
e5-base-v2	768	0.67
bge-base-en-v1.5	768	0.67
granite-embedding-30m-english	384	0.16
granite-embedding-125m-english	768	0.64
multilingual-e5-small	384	0.30
multilingual-e5-base	768	0.67
granite-embedding-107m-multilingual	384	0.17
granite-embedding-278m-multilingual	768	0.67

Table 8: Retrieval Time per Query

8 Conclusion

We present the Granite Embedding Models, a family of open, lightweight embedding models curated for enterprise retrieval tasks. We release 5 models spanning two model sizes for a variety of latency budgets. We introduce English models that can be used in both dense and sparse vector applications, and multilingual models for dense vector search applications in 12 languages. We provide descriptions of the training data and training mechanism, with a focus on high quality data appropriate for enterprise use. We show our models achieve higher scores in average on a variety of text embeddings and retrieval benchmarks, and show how our lightweight models can be used for applications which require very low retrieval times. All out models are released under an Apache 2.0 license, allowing for both commercial and research purposes. We plan to continuously release updates to these models to improve their performance and introduce relevant features.

A Contributions

Core Model Training

Parul Awasthy, Radu Florian, Yulong Li, Scott McCarley, Arafat Sultan, Aashka Trivedi

Data and Evaluation

Parul Awasthy, Mihaela Bornea, Radu Florian, Martin Franz, Bhavani Iyer, Vishwajeet Kumar, Rudra Murthy, Sara Rosenthal, Avirup Sil, Arafat Sultan, Aashka Trivedi, Yulong Li

Retrieval Oriented Pretraining

Vishwajeet Kumar, Rudra Murthy, Vignesh P, Aashka Trivedi

Product Management

Abraham Daniels, Gabe Goodhart, Sukriti Sharma, Kate Soule

Technical Leadership

David Cox, Radu Florian, Luis Lastras, Salim Roukos, Jaydeep Sen

B English Text Embedding Evaluation

We show the performance of the Granite Embedding models on the English MTEB benchmark Muennighoff et al. (2022) in Table 9. The MTEB benchmarks spans 8 tasks and 56 datasets, used to evaluate the quality of text embeddings for classification, clustering, pair classification, re-ranking, retrieval, re-ranking, sentence similarity and summarization tasks.

We provide the average score per task, using the main metric described in the paper: Accuracy for classification tasks, V-Measure for clustering tasks, Average Precision for pair classification tasks, MAP for re-ranking tasks, nDCG@10 for retrieval tasks and Spearman Correlation (based on cosine similarity) for STS and summarization tasks.

Model	Classification	Clustering	Pair Classification	Re-ranking	Retrieval	STS	Summarization	**Avg.**
e5-small-v2	72.9	39.9	84.7	54.3	49.0	80.4	31.2	59.9
bge-small-en-v1.5	74.1	43.8	84.9	58.4	51.7	81.6	30.1	62.2
e5-base-v2	73.8	44.1	85.7	55.9	50.3	81.1	30.3	61.6
bge-base-en-v1.5	75.5	45.8	86.6	58.9	53.3	82.4	31.1	63.6
granite-embedding-30m-english	61.2	43.2	78.9	58.5	49.1	76.2	29.4	57.3
granite-embedding-125m-english	62.0	45.3	79.6	60.3	52.3	73.1	31.8	58.4
multilingual-e5-small	67.0	37.2	82.7	52.9	46.6	79.2	30.0	57.1
multilingual-e5-base	69.8	39.6	83.6	55.0	48.9	80.3	30.2	59.1
granite-embedding-107m-multilingual	62.1	37.8	80.5	55.6	45.3	71.6	31.1	54.5
granite-embedding-278m-multilingual	63.5	39.2	81.4	56.1	48.2	72.9	32.0	56.1

Table 9: English MTEB Benchmark Muennighoff et al. (2022) scores.

C Multilingual Text Retrieval Evaluation

To evaluate our multilingual embedding models on out-of-domain information retrieval, we use multilingual retrieval tasks from the MTEB repository. While MTEB provides an official benchmark for Polish, French, and Chinese, we pick the available retrieval tasks from German (Table 10), Spanish (Table 11), French (Table 12), Japanese (Table 13), Arabic (Table 14), Korean (Table 15) and Chinese (Table 16). For English, we evaluate on the BEIR benchmark in Table 2

Model	QuAD-Retrieval	DPR	GovService-Retrieval	LegalQuAD	Avg.
multilingual-e5-small	93.14	78.94	82.78	47.8	75.7
multilingual-e5-base	93.93	79.51	85.07	47.85	76.6
granite-embedding-107m-multilingual	90.7	77.2	80.3	32.8	70.3
granite-embedding-278m-multilingual	91.3	78.4	78.1	36.8	71.2

Table 10: German Retrieval nDCG@10 scores

Model	Passage Retrieval S2P	Passage Retrieval S2S	**Avg.**
multilingual-e5-small	38.99	62.99	51.0
multilingual-e5-base	39.14	65.52	52.3
granite-embedding-107m-multilingual	35.7	61.7	48.7
granite-embedding-278m-multilingual	39.1	66.1	52.6

Table 11: Spanish Retrieval nDCG@10 scores

Model	Alloprof Retrieval	BSARD Retrieval	Syntec Retrieval	FQuAD Retrieval	XPQA Retrieval	**Avg.**
multilingual-e5-small	27.38	14.54	73.46	78.78	57.17	50.3
multilingual-e5-base	34.44	18.83	82.86	82.48	59.52	55.6
granite-embedding-107m-multilingual	38.9	14.9	79.7	69	53	51.1
granite-embedding-278m-multilingual	44.7	16.4	81.1	70.1	58.2	54.1

Table 12: French Retrieval nDCG@10 scores

Model	JaQuAD Retrieval	JaGovFaq Retrieval	**Avg.**
multilingual-e5-small	59	65.6	62.3
multilingual-e5-base	61.0	68	64.5
granite-embedding-107m-multilingual	56.4	61.5	59
granite-embedding-278m-multilingual	57.3	66.1	61.7

Table 13: Japanese Retrieval nDCG@10 scores

Model	Saddeem Questions
multilingual-e5-base	69.7
multilingual-e5-small	68.6
granite-embedding-278m-multilingual	64.2
granite-embedding-107m-multilingual	63.2

Table 14: Arabic Retrieval nDCG@10 scores

Model	Ko-Starategy Qa
multilingual-e5-small	75.1
multilingual-e5-base	76.4
granite-embedding-107m-multilingual	70.5
granite-embedding-278m-multilingual	71.8

Table 15: Korean Retrieval nDCG@10 scores

Model	CMedQa Retrieval	CovidRetrieval	DuRetrieval	EcomRetrieval	MedicalRetrieval	MSMarcp	T2Retrieval	VideoRetrieval	**Avg.**
multilingual-e5-small	24.4	72.8	81.4	53.5	44.8	73.2	71.4	58	59.9
multilingual-e5-base	27.2	73.5	81.6	54	48.3	76	70.8	61.2	61.6
granite-embedding-107m-multilingual	16.1	59.7	57.6	22.1	27.3	58.5	57.7	27.4	40.8
granite-embedding-278m-multilingual	19.6	58	64.1	28.1	34	62.5	63.9	31.6	45.2

Table 16: Chinese Retrieval nDCG@10 scores

REFERENCES

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone, 2024. URL https://arxiv.org/abs/2404.14219.

Dimo Angelov. Top2vec: Distributed representations of topics. *CoRR*, abs/2008.09470, 2020. URL https://arxiv.org/abs/2008.09470.

Yang Bai, Xiaoguang Li, Gang Wang, Chaoliang Zhang, Lifeng Shang, Jun Xu, Zhaowei Wang, Fangshan Wang, and Qun Liu. Sparterm: Learning term-based sparse representation for fast text retrieval, 2020. URL https://arxiv.org/abs/2010.00768.

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, 2018. URL https://arxiv.org/abs/1611.09268.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470, 2020. doi: 10.1162/tacl_a_00317. URL https://aclanthology.org/2020.tacl-1.30.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. SPECTER: Document-level Representation Learning using Citation-informed Transformers. In *ACL*, 2020.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.747. URL https://aclanthology.org/2020.acl-main.747.

Zhuyun Dai and Jamie Callan. Context-aware sentence/passage term importance estimation for first stage retrieval, 2019. URL https://arxiv.org/abs/1910.10687.

Matthew Dunn, Levent Sagun, Mike Higgins, V. Ugur Guney, Volkan Cirik, and Kyunghyun Cho. Searchqa: A new q&a dataset augmented with context from a search engine, 2017.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pp. 1156–1165, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450329569. doi: 10.1145/2623330.2623677. URL https://doi.org/10.1145/2623330.2623677.

Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. Splade v2: Sparse lexical and expansion model for information retrieval, 2021a. URL https://arxiv.org/abs/2109.10086.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. Splade: Sparse lexical and expansion model for first stage ranking, 2021b. URL https://arxiv.org/abs/2107.05720.

Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *International Conference on Machine Learning*, 2018.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6894–6910, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.552. URL https://aclanthology.org/2021.emnlp-main.552.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network. In *NeurIPS Deep Learning Worksop*, 2014.

Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. Improving efficient neural ranking models with cross-architecture knowledge distillation, 2021. URL https://arxiv.org/abs/2010.02666.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts, 2024. URL https://arxiv.org/abs/2401.04088.

Kyungyul Kim, ByeongMoon Ji, Doyoung Yoon, and Sangheum Hwang. Self-knowledge distillation with progressive refinement of targets. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6547–6556. IEEE, October 2021. doi: 10.1109/iccv48922.2021.00650. URL http://dx.doi.org/10.1109/ICCV48922.2021.00650.

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. Natural Questions: A Benchmark for Question Answering Research. *Transactions of the ACL*, 2019.

Carlos Lassance, Hervé Déjean, Thibault Formal, and Stéphane Clinchant. Splade-v3: New baselines for splade, 2024. URL https://arxiv.org/abs/2403.06789.

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, 2024.

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. PAQ: 65 Million Probably-Asked Questions and What You Can Do With Them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115, 10 2021. ISSN 2307-387X. doi: 10.1162/tacl_a_00415. URL https://doi.org/10.1162/tacl_a_00415.

Xiangyang Li, Kuicai Dong, Yi Quan Lee, Wei Xia, Yichun Yin, Hao Zhang, Yong Liu, Yasheng Wang, and Ruiming Tang. Coir: A comprehensive benchmark for code information retrieval models, 2024. URL https://arxiv.org/abs/2407.02883.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning, 2023. URL https://arxiv.org/abs/2308.03281.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. URL https://arxiv.org/abs/1907.11692.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. S2ORC: The semantic scholar open research corpus. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4969–4983, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.447. URL https://aclanthology.org/2020.acl-main.447.

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

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022. doi: 10.48550/ARXIV.2210.07316. URL https://arxiv.org/abs/2210.07316.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, Johannes Heidecke, Pranav Shyam, Boris Power, Tyna Eloundou Nekoul, Girish Sastry, Gretchen Krueger, David Schnurr, Felipe Petroski Such, Kenny Hsu, Madeleine Thompson, Tabarak Khan, Toki Sherbakov, Joanne Jang, Peter Welinder, and Lilian Weng. Text and code embeddings by contrastive pre-training, 2022. URL https://arxiv.org/abs/2201.10005.

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

Biswajit Paria, Chih-Kuan Yeh, Ian E. H. Yen, Ning Xu, Pradeep Ravikumar, and Barnabás Póczos. Minimizing flops to learn efficient sparse representations, 2020. URL https://arxiv.org/abs/2004.05665.

Yifu Qiu, Hongyu Li, Yingqi Qu, Ying Chen, Qiaoqiao She, Jing Liu, Hua Wu, and Haifeng Wang. Dureader retrieval: A large-scale chinese benchmark for passage retrieval from web search engine, 2022.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for squad. *CoRR*, abs/1806.03822, 2018. URL http://arxiv.org/abs/1806.03822.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410.

Sara Rosenthal, Avirup Sil, Radu Florian, and Salim Roukos. ¡scp¿clapnq¡/scp¿: ¡u¿c¡/u¿ohesive ¡u¿l¡/u¿ong-form ¡u¿a¡/u¿nswers from ¡u¿p¡/u¿assages in natural questions for rag systems. *Transactions of the Association for Computational Linguistics*, 13:53–72, 2025. ISSN 2307-387X. doi: 10.1162/tacl_a_00729. URL http://dx.doi.org/10.1162/tacl_a_00729.

SadeemAI. Sadeem qna: An arabic qna dataset. https://huggingface.co/datasets/sadeem-ai/arabic-qna, 2024.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 1604–1619, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.138.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune bert for text classification? In *China national conference on Chinese computational linguistics*, pp. 194–206. Springer, 2019.

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 Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *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, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1074. URL https://aclanthology.org/N18-1074.

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, 2023.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models, 2024a. URL https://arxiv.org/abs/2401.00368.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Multilingual e5 text embeddings: A technical report, 2024b.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers, 2021. URL https://arxiv.org/abs/2012.15828.

Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. RetroMAE: Pre-training retrieval-oriented language models via masked auto-encoder. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 538–548, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.35. URL https://aclanthology.org/2022.emnlp-main.35.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-pack: Packaged resources to advance general chinese embedding, 2023a.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Xingrun Xing. Lm-cocktail: Resilient tuning of language models via model merging, 2023b. URL https://arxiv.org/abs/2311.13534.

Xiaohui Xie, Qian Dong, Bingning Wang, Feiyang Lv, Ting Yao, Weinan Gan, Zhijing Wu, Xiangsheng Li, Haitao Li, Yiqun Liu, and Jin Ma. T2ranking: A large-scale chinese benchmark for passage ranking. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, pp. 2681–2690. ACM, July 2023. doi: 10.1145/3539618.3591874. URL http://dx.doi.org/10.1145/3539618.3591874.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval, 2020. URL https://arxiv.org/abs/2007.00808.

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 Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259.

Hamed Zamani, Mostafa Dehghani, W. Bruce Croft, Erik Learned-Miller, and Jaap Kamps. From neural re-ranking to neural ranking: Learning a sparse representation for inverted indexing. In

Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM '18, pp. 497–506, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450360142. doi: 10.1145/3269206.3271800. URL https://doi.org/10.1145/3269206.3271800.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval, 2024. URL https://arxiv.org/abs/2407.19669.

Xinyu Zhang, Nandan Thakur, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Mehdi Rezagholizadeh, and Jimmy Lin. MIRACL: A multilingual retrieval dataset covering 18 diverse languages. *Transactions of the Association for Computational Linguistics*, 11:1114–1131, 2023. doi: 10.1162/tacl_a_00595. URL https://aclanthology.org/2023.tacl-1.63.

Tiancheng Zhao, Xiaopeng Lu, and Kyusong Lee. Sparta: Efficient open-domain question answering via sparse transformer matching retrieval, 2020. URL https://arxiv.org/abs/2009.13013.