Unsupervised Adaptation of Large Language Models for Dense Retrieval

Anonymous ACL submission

Abstract

Dense retrieval calls for discriminative embeddings to represent the semantic relationship between query and document. It may benefit from the using of large language models (LLMs), given LLMs' strong capability on semantic understanding. However, the LLMs are learned by auto-regression, whose working mechanism is completely different from representing whole text as one discriminative embedding. Thus, it is imperative to study how to adapt LLMs properly so that they can be effectively initialized as the backbone encoder for dense retrieval.

In this paper, we propose a novel approach, called **LLaRA** (LLM adatpeted for dense RetrivAl), which performs unsupervised adaptation of LLM for its dense retrieval application. LLaRA consists of two pretext tasks: EBAE (Embedding-Based Auto-Encoding) and EBAR (Embedding-Based Auto-Regression), where the LLM is prompted to *reconstruct the input sentence* and *predict the next sentence* based on its text embeddings. LLaRA is simple, lightweight, but highly effective. It is used to adapt LLaMA-2-7B on the Wikipedia corpus. With a moderate steps of adaptation, it substantially improves the model's fine-tuned performances on on a variety of dense retrieval benchmarks. Notably, it results in the new state-of-the-art performances on popular benchmarks, such as passage and document retrieval on MS-MARCO, and zero-shot retrieval on BEIR. The model and source code will be made publicly available to facilitate the future research.

1 Introduction

Dense retrieval is a new paradigm of IR empowered by deep neural networks. It represents query and document as embeddings within the same latent space, where the semantic relationship between query and document can be reflected by their embedding similarity. Nowadays, dense retrieval has been a critical component in many real-world scenarios, like open-domain QA, fact verification, and retrieval-augmented generation (Karpukhin et al., 2020; Thorne et al., 2018; Lewis et al., 2020).

The capacity of text encoder is a critical factor of dense retrieval. In the past few years, the pre-trained language models, e.g., BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), T5 (Raffel et al., 2020), were widely applied to generate high-quality representations for query and document, which substantially contributed to the accuracy and generalizability of dense retrieval. Besides, it was also found that dense retrieval's performance can further benefit from the continual growth of model size and training scale (Ni et al., 2021; Izacard et al., 2021; Wang et al., 2022b; Xiao et al., 2023). Recently, large language models (LLMs) have emerged as a unified foundation for general NLP tasks (Brown et al., 2020; Wei et al., 2021; Chowdhery et al., 2023). Given the LLMs' superior capability on semantic understanding, it will be promising to take advantage of such powerful models as new backbones for dense retrieval. With this consideration, there have been pioneering efforts where LLMs are trained to generate discriminative text embeddings to facilitate the retrieval tasks in many different scenarios (Muennighoff, 2022; Neelakantan et al., 2022; Ma et al., 2023; Zhang et al., 2023).

Despite the preliminary progress, it remains an open problem to fully unleash the LLM's underlying potential for dense retrieval. Particularly, the typical LLMs are pre-trained by text generation tasks, especially auto-regression, where the text embeddings are learned to predict the next tokens. Consequently, the LLMs' text embeddings will focus on representing the local (i.e. the next step) semantic of the context. However, dense retrieval calls for text embeddings to represent the global semantic about the query and document. Such a big discrepancy will severely restrict the direct application of LLMs for dense retrieval.

To address the above problem, we propose a novel approach **LLaRA** (Figure 1), which per-

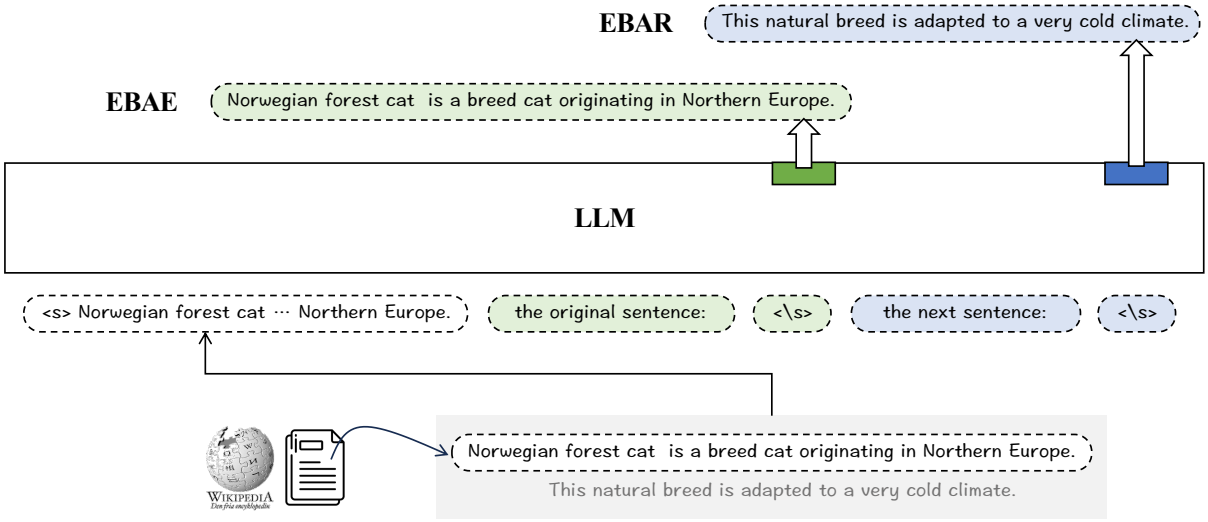

Figure 1: **LLaRA**. The LLM is prompted to generate the inductive embedding for EBAE (green), where the original sentence is predicted, and the deductive embedding for EBAR (blue), where the next sentence is predicted.

forms unsupervised adaptation of LLMs to facilitate their usage in dense retrieval. LLaRA works as a continuation of pre-training. On top of the tailored unsupervised learning tasks, it transforms the LLM's text embeddings to represent the global semantic about the input text, which makes the LLM a better initialized encoder for dense retrieval. LLaRA is made up of two pretext training tasks: **EBAE** (embedding-based auto-encoding) and **EBAR** (embedding-based auto-regression). On top of EBAE, the LLM is prompted to generate the inductive text embedding, which can be used to predict the vocabularies of the input sentence itself. With EBAR, the LLM is prompted to generate the deductive text embedding, which can be used to predict the vocabularies for the next sentence.

The joint conduct of the above pretext tasks bring forth two benefits. Firstly, the text embeddings from LLM can be adapted from *local* semantic representation (predicting the next tokens) to *global* semantic representation (predicting the sentence-level features), which aligns with the expected property of dense retrieval. Secondly, by learning to generate inductive and deductive text embeddings with different prompts, the LLM-based retriever can flexibly handle diversified semantic relationships about correlation (e.g., QA) and paraphrasing (e.g., NLI), which presents a strong foundation to develop versatile retrieval models. It's worth noting that the prediction is realized in the form of multi-classification, where the LLM's text embedding is the input and the vocabularies within the target sentence are employed as the labels. Therefore, LLaRA is extremely lightweight and simple to realize based on the existing auto-regression pipeline.

We apply LLaRA for LLaMA-2-7B (base) (Touvron et al., 2023) over the Wikipedia corpus, where it substantially improves the LLM's downstream retrieval performance. With standard fine-tuning, the well-adapted model is able to notably outperform all existing dense retrieval methods, where it establishes new state-of-the-art performances on a variety of popular benchmarks, including the supervised tasks like passage and document retrieval of MSMARCO (Nguyen et al., 2016), and the zero-shot retrieval of BEIR (Thakur et al., 2021).

To summarize, our work is highlighted by the following technical contributions. 1) We propose a new unsupervised learning method LLaRA. To the best of our knowledge, this is the first research work which explores the adaptation of LLMs for dense retrieval. 2) LLaRA is designed with simple but effective pretext tasks, which substantially improves the quality of LLM-based dense retriever in a cost-effective way. 3) The empirical studies verify the effectiveness of LLaRA, where substantial improvements can be achieved for both supervised and zero-shot retrieval tasks.

2 Related Works

In this section, the related works are discussed from two perspectives: the background of dense retrieval, and the previous efforts on leveraging the LLMs for dense retrieval applications.

• **Dense retrieval**. Dense retrieval is to represent query and document as embeddings within the same latent space, where relevant documents can be retrieved by embedding similarity. Nowadays, it

is widely utilized in many important applications, such as open-domain QA and retrieval-augmented generation (Karpukhin et al., 2020; Lewis et al., 2020). The performance of dense retrieval is influenced by many factors. For example, dense retrieval models are learned by contrastive learning, where the discriminativeness of text embeddings are largely influenced by the scale and hardness of negative sample (Qu et al., 2020; Izacard et al., 2021; Xiong et al., 2020). Besides, the learning of dense retrieval models can benefit from knowledge distillation, where fine-grained teacher labels are derived from the ranking models (Hofstätter et al., 2021; Ren et al., 2021). Apart from the above training algorithms, the backbone architecture is one decisive factor for dense retrieval. In the past few years, pre-trained language models (PLMs), like BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), T5 (Raffel et al., 2020), have been widely adopted for the encoding of query and documents. Thanks to the large-scale model architecture and pre-training, PLMs were able to produce fine-grained semantic representation of input data, which substantially benefit the quality of dense retrieval. Besides, it was found that with the expansion of model and training scale, and the optimization of pre-training algorithm, the accuracy and generality of the PLM-based dense retrieval can be further improved. (Ni et al., 2021; Izacard et al., 2021; Wang et al., 2022b; Xiao et al., 2023; Gao and Callan, 2021; Liu and Shao, 2022; Liu et al., 2023; Wang et al., 2022a).

• **Dense retrieval with LLM**. The LLMs have been a unified foundation for many NLP tasks because of its superior capabilities. As a result, it is instinctive to leverage such powerful models to facilitate dense retrieval. LLMs can substantially contribute to many critical aspects of dense retrieval. For example, it can help to model the complex relationship between query and document considering LLMs' strong semantic understanding capability (Brown et al., 2020; Chowdhery et al., 2023; Touvron et al., 2023). Besides, it will benefit the learning of multi-task retrievers because of LLMs' versatility and instruction following capability (Wei et al., 2021; Chung et al., 2022). It also presents a powerful foundation to develop long-document retrievers, given its dramatically extended context lengths. Recently, there have been several preliminary works which made important progresses on LLM-based dense retrieval (Muennighoff, 2022; Neelakantan et al., 2022; Ma et al., 2023; Zhang et al., 2023). However, the existing methods simply made direct use of LLMs. Because of the discrepancy between language modeling and text embedding, much of the LLMs' underlying potential is unexploited. In fact, it is still an open problem to study the proper adaptation of LLMs so that they can better contribute to dense retrieval.

3 LLaRA

3.1 Preliminary

Dense retrieval utilizes a text embedding model to produce the query and document's embedding: $\boldsymbol{e}_q$ and $\boldsymbol{e}_d$. The relevance of query and document is reflected by their embedding similarity: $\langle \boldsymbol{e}_q, \boldsymbol{e}_d \rangle$. As such, the relevant documents for the query (D_q) can be retrieved via the ANN search within the embedding space: $D_q \leftarrow \text{Top-}k(\{d : \langle \boldsymbol{e}_q, \boldsymbol{e}_d \rangle | D\})$.

The pre-trained language models used to be the backbone architecture of the embedding model. Take BERT as an example. The input text is tokenized as the sequence T: [CLS], t1, ..., tN, [EOS]. Then, the tokenized sequence is encoded by BERT, where the output embeddings are integrated as the text embedding. There are two common options to perform the integration: [CLS], or mean-pooling:

$$\boldsymbol{e}_t \leftarrow \text{BERT}(T)[\text{CLS}] \; \textit{or} \; \text{AVG}\big(\text{BERT}(T)\big).$$

When using LLMs as the encoding architecture, the text embedding needs to be generated in a different way. Since the existing LLMs mainly use the decoder-only architecture (Brown et al., 2020; Chowdhery et al., 2023; Touvron et al., 2023), the global context can only be accessed by the very last tokens of the input sequence. Therefore, the output embedding from the special token $\langle \backslash \text{s} \rangle$ or [EOS] is utilized to represent the input text (Zhang et al., 2023; Ma et al., 2023). Taking LLaMA (Touvron et al., 2023) as the example, we have the following updated form of text embedding:

$$\boldsymbol{e}_t \leftarrow \text{LLaMA}(T)[\langle \backslash \text{s} \rangle]. \quad (1)$$

3.2 Unsupervised Adaptation

As introduced, the LLM's output embedding tends to foucs on local semantic because of its language modeling-based pre-training. To facilitate the application in dense retrieval, we perform unsupervised adaptation of LLM, where the LLM's text embedding can be transformed into a semantic representation of the global context.

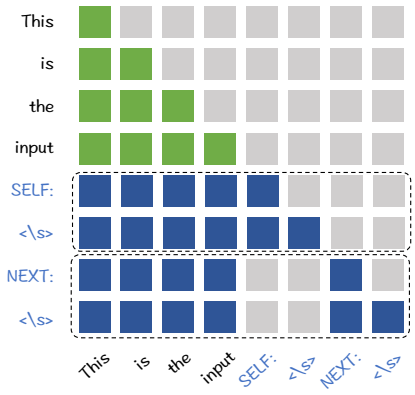

Figure 2: The attention scheme of LLaRA.

• **Objective**. The retrieval tasks can be roughly divided into two types. One is to find the correlated data, e.g., question-answering (QA). The other one is to identify paraphrasing data, e.g., natural language inference (NLI). To confront these tasks, text embeddings are expected to fulfill two objectives:

- *Induction*. The representation of global semantic about the input text itself.
- *Deduction*. The representation of global semantic for the correlated text of the input.

• **Pretext tasks**. With the above objectives, we present two pretext training tasks of LLaRA. One is called **EBAE** (Embedding-based Auto-Encoding), where the text embedding $\boldsymbol{e}_t$ is used to induct the sentence-level feature about input sentence itself. The other one is called **EBAR** (Embedding-based Auto-Regression), where the text embedding $\boldsymbol{e}_t$ is used to deduct the sentence-level feature for the next sentence of the input. We argue that the strong induction and deduction capability of text embedding will be sufficient to handle the diversified retrieval scenarios, considering that arbitrary correlations can always be abstracted into the general form of *input* (e.g., question), *retrieval purpose* (e.g., get its answer), *next sentence* (e.g, answer).

• **Text Embedding**. The LLM is prompted by two different templates to generate the text embeddings for EBAE and EBAR (Figure 1). For EBAE, the LLM is prompted by: "[Placeholder for input]␣SELF␣$\langle\backslash$s$\rangle$", where the inductive text embedding is generated by the following function:

$$\boldsymbol{e}_t^{\alpha} \leftarrow \text{LLaMA}(T, \text{SELF}, \langle\backslash\text{s}\rangle)[-1]. \quad (2)$$

SELF stands for the prompt of EBAE: "The input sentence is:". For EBAR, the LLM is prompted by template: "[Placeholder for input]␣NEXT␣$\langle\backslash$s$\rangle$", based on which the deductive text embedding is generated by the following function:

$$\boldsymbol{e}_t^{\beta} \leftarrow \text{LLaMA}(T, \text{NEXT}, \langle\backslash\text{s}\rangle)[-1], \quad (3)$$

In this place, NEXT stands for the prompt of EBAR: "The next sentence is:".

The direct computation of the two embeddings will lead to substantial unnecessary costs because the input text T is repetitively processed for two times. To alleviate this problem, we propose to compute $\boldsymbol{e}_t^{\alpha}$ and $\boldsymbol{e}_t^{\beta}$ in one pass. Particularly, the prompts of EBAE and EBAR are merged into one joint prompt for LLM: "[Placeholder for input]␣SELF␣$\langle\backslash$s$\rangle$␣NEXT␣$\langle\backslash$s$\rangle$". Because the two text embeddings need to be computed independently, we modify the typical attention mask of casual language modeling, where "SELF␣$\langle\backslash$s$\rangle$" and "NEXT␣$\langle\backslash$s$\rangle$" are mutually invisible (Figure 2). Now, the output embeddings of the first and second $\langle\backslash$s$\rangle$ tokens are used for $\boldsymbol{e}_t^{\alpha}$ and $\boldsymbol{e}_t^{\beta}$, respectively. Given that the input text T account for the majority of length for the joint prompt, such an operation will save almost 50% of the computation cost compared with the naive method.

• **Training**. As introduced, the text embeddings of LLaRA are adapted to capture the global semantic of the input sentence itself and the next sentence of the input. In this place, we propose a simple but effective training objective to accomplish such an adaptation. We argue that *if one embedding is able to accurately predict the vocabularies in a specific context all by itself*, the embedding must be a strong representation of the global semantic for the corresponding context. Based on this fundamental principle, the training of text embedding is formulated as a multi-classification problem. It linearly projects the text embedding into the vocabulary space, where the vocabulary IDs of all tokens within the target context are predicted. Specifically, the objective function of this problem is derived as:

$$\min . -\frac{1}{|T|}\sum_{t\in\mathcal{T}} log\frac{\exp\left(\boldsymbol{e}^T\boldsymbol{W}_t\right)}{\sum_{v\in V}\exp\left(\boldsymbol{e}^T\boldsymbol{W}_v\right)}. \quad (4)$$

In this place, $\boldsymbol{W} \in \mathbb{R}^{d\times|V|}$ is the projection head of LLM; V indicates the vocabulary space; $\mathcal{T}$ stands for the collection of tokens of the target context (input text itself for $\boldsymbol{e}_t^{\alpha}$, the next sentence for $\boldsymbol{e}_t^{\beta}$). The above training objective is lightweight and simple to realize, which can be directly conducted based on the typical language modeling pipeline.

-			Dev		DL'19	DL'20
Method	**Size**	**FT.**	**M@10**	**R@1000**	**N@10**	**N@10**
BM25 (Lin et al., 2021)	–	–	18.4	85.3	50.6	48.0
ANCE (Xiong et al., 2020)	125M	hard	33.0	95.9	64.8	–
ADORE (Zhan et al., 2021)	110M	hard	34.7	–	68.3	–
Condenser (Gao and Callan, 2021)	110M	hard	36.6	97.4	69.8	–
coCondenser (Gao and Callan, 2022)	110M	hard	38.2	98.4	71.7	68.4
TAS-B (Hofstätter et al., 2021)	55M	distill	34.3	97.6	72.2	69.2
RocketQAv2 (Ren et al., 2021)	–	distill	38.8	98.1	–	–
AR2+SimANS (Zhou et al., 2022)	110M	distill	40.9	98.7	–	–
GTR-XXL (Ni et al., 2021)	4.8B	–	38.8	99.0	–	–
SimLM (Wang et al., 2022a)	110M	hard	39.1	98.6	69.8	69.2
SimLM+distill (Wang et al., 2022a)	110M	distill	41.1	98.7	71.4	69.7
RetroMAE (Liu and Shao, 2022)	110M	hard	39.3	98.5	–	–
RetroMAE+distill (Liu and Shao, 2022)	110M	distill	41.6	98.8	68.1	–
RetroMAEv2+distll (Liu et al., 2023)	110M	distill	42.6	98.9	**75.1**	–
LLaMA2-RepLLaMA (Ma et al., 2023)	7B	hard	41.2	99.4	74.3	72.1
OpenAI-Ada-002 (Neelakantan et al., 2022)	–	–	34.4	98.6	70.4	67.6
LLaMA2-LLaRA	7B	hard	**43.1**	**99.5**	73.4	**72.9**

Table 1: MS MARCO passage retrieval (performance measured by MRR@10, Recall@1000, NDCG@10).

• **Fine-Tuning**. The well-adapted LLM from LLaRA is fine-tuned for dense retrieval applications through contrastive learning. Because the majority of the fine-tuning datasets for dense retrieval are collected for correlation scenarios, such as QA (Nguyen et al., 2016) and Natural Question (Kwiatkowski et al., 2019), which are made up of tuples of (query, answer), we can derive the following general form of objective function:

$$\min \sum_{q} -\log \frac{\exp(\langle \boldsymbol{e}_q^{\alpha}, \boldsymbol{e}_a^{\beta} \rangle)}{\sum_{a' \in \mathcal{A}'} \exp(\langle \boldsymbol{e}_q^{\alpha}, \boldsymbol{e}_{a'}^{\beta} \rangle)}, \quad (5)$$

where $\boldsymbol{e}_q^{\alpha}$ is the query's embedding prompted by NEXT, and $\boldsymbol{e}_a^{\beta}$ is the answer's embedding prompted by SELF. Despite the fixed formulation during training, the prompt scheme can be flexibility adjusted for each individual downstream scenario. Particularly, when dealing with the correlation relationships, e.g., question-answering, we hold on to NEXT and SELF to prompt the query and answer's embeddings. However, when handling other situations about paraphrasing relationships, the prompt scheme is as follows. To analyze the paraphrasing relationship between two long documents, we use SELF to prompt the query and answer's embeddings given its nature of summarizing the semantic of complex input. Meanwhile, we employ NEXT as the prompt for both inputs when dealing with two short sentences because of its nature of deducting the semantic for the related texts.

4 Experiment

4.1 Settings

The experimental study is performed to explore three important issues: 1) LLaRA's retrieval performance after fine-tuning, 2) LLaRA's generalization across diversified scenarios, 3) the impact of technical factors in LLaRA. With such objectives, we use the MS MARCO (Nguyen et al., 2016) as our fine-tuning dataset, and perform the evaluation on the passage retrieval and document retrieval task. We also take advantage of the BEIR benchmark (Thakur et al., 2021), where the fine-tuned retriever from MS MARCO is evaluated under the zero-shot setting to analyze its generalization capability.

• **Training**. LLaRA is applied to the LLaMA-2-7B (base) model. The unsupervised adaptation is performed based on the unlabeled corpus of Wikipedia curated by DPR (Karpukhin et al., 2020). We perform 10,000 steps of LLaRA adaptation in total, with a batch size of 256, a sequence length of 1024, and a learning rate of 1e-5. LLaRA is fine-tuned based on the training recipe presented by RepLLaMA (Ma et al., 2023): it leverages LoRA (Hu et al., 2021) for the parameter efficient training of LLM, and simply relies on the ANN hard negatives (Xiong et al., 2020) to fine-tune the embedding model with contrastive learning.

4.2 Supervised Performance

First of all, we analyze the supervised retrieval quality of LLaRA, where the model is fine-tuned with

-			Dev		DL'19	DL'20
Method	**Size**	**FT.**	**MRR@100**	**R@100**	**NDCG@10**	**NDCG@10**
BM25 (Lin et al., 2021)	–	–	27.7	80.9	51.9	52.9
PROP (Ma et al., 2021a)	110M	–	39.4	88.4	59.6	–
B-PROP (Ma et al., 2021b)	110M	–	39.5	88.3	60.1	–
COIL (Gao et al., 2021)	110M	hard	39.7	–	63.6	–
ANCE (first-p) (Xiong et al., 2020)	125M	hard	37.7	89.3	61.5	–
ANCE (max-p) (Xiong et al., 2020)	125M	hard	38.4	90.6	62.8	–
ADORE (Zhan et al., 2021)	110M	hard	40.5	91.9	62.8	–
COSTA (Ma et al., 2022)	110M	hard	42.2	91.9	62.6	–
LLaMA2-RepLLaMA (Ma et al., 2023)	7B	hard	45.6	–	65.0	63.2
LLaMA2-LLaRA	7B	hard	**47.9**	**94.1**	**68.2**	**63.6**

Table 2: MS MARCO document retrieval

the training queries from MS MARCO passage and document retrieval, respectively.

• **Passage Retrieval**. The experiment results on MS MARCO passage retrieval are shown in Table 1. We make comparison with a wide variety of baseline methods on passage retrieval, which include the following categories: 1) basic lexical retriever: BM25 (Lin et al., 2021); 2) dense retrievers fine-tuned from BERT or RoBERTa: ANCE (Xiong et al., 2020), ADORE (Zhan et al., 2021), AR2+SimANS (Zhou et al., 2022), RocketQAv2 (Ren et al., 2021); 3) dense retrievers fine-tuned from the enhanced PLMs: Condenser (Gao and Callan, 2021), coCondenser (Gao and Callan, 2022), RetroMAE (Liu and Shao, 2022; Liu et al., 2023), SimLM (Wang et al., 2022a); 4) dense retrievers based on LLMs: GTR-XXL based on T5-4.8B (Ni et al., 2021), SGPT (Muennighoff, 2022) and OpenAI-Ada-002 (Neelakantan et al., 2022) based on GPT, RepLLaMA (Ma et al., 2023) based on LLaMA-2-7B. RepLLaMA is the closest baseline to our method, which directly fine-tunes the original LLaMA-2-7B backbone without any adaptation. There are two different fine-tuning methods (FT.): one is based on hard-negative sampling (hard): which is simple and low-cost; the other one is based on knowledge distillation (distill), which is accurate but expensive due to its demand of a precise ranker and complicated training process.

The primary observations are presented as follows. First of all, LLaRA achieves a superior retrieval performance in every evaluation scenario. Remarkably, it achieves a MRR@10 of **43.1** and a Recall@1000 of **99.5**, which notably improves the performance of the baselines and presents a new state-of-the-art result on the large-scale dev set. Its performance is also highly competitive on DL'19 and DL'20, though it's slightly lower on DL'19 due to the randomness of the small test set. Besides, it leads to a notably improvement over the closest baseline RepLLaMA (based on the same backbone but without adaptation), which indicates the effect introduced by the adaptation of LLaRA. Finally, we can observe the the LLM-based retrievers' overwhelming advantages in comparison with the previous ones based on smaller PLMs, despite that the they are usually fine-tuned with a relatively simple approach (hard). Compared with the best PLM baseline fine-tuned by hard negatives, RetroMAE+hard and SimLM+hard, the switch to LLaRA brings forth almost +4% gains in MRR@10. Such a dramatic improvement validates the LLMs' huge potential for dense retrieval, and with proper adaptation, such a potential can be exploited more effectively.

• **Document Retrieval**. We report the evaluation results on MS MARCO document retrieval in Table 2. We make comparison with popular document retrieval methods, including BM25 (Lin et al., 2021), ADORE (Zhan et al., 2021), ANCE first-p and max-p (Xiong et al., 2020), PROP (Ma et al., 2021a), B-PROP (Ma et al., 2021b), COIL (Gao et al., 2021), COSTA (Ma et al., 2022), RepLLaMA (Ma et al., 2023), which fall into the same categories as the passage retrievers.

Our observations on document retrieval is very similar with our previous result on passage retrieval. In particular, LLaRA achieves a superior empirical performance in every evaluation, where it notably improves the previous BERT-based methods by **+5.7%** point in MRR@100. Both LLM-based retrievers, RepLLaMA and LLaRA, are able to dominate the PLM-based baselines. Besides, LLaRA continues to outperform RepLLaMA, with a **+2.3%** improvement in MRR@100 on the large-scale dev set and consistent advantages on DL'19 and DL'20.

Method	BM25	BERT	GTR-XXL	CPT-XL	Ada-2	SGPT	RepLLaMA	LLaRA
Size	–	110M	4.8B	175B	–	5.8B	7B	7B
T-COVID	59.5	61.5	50.1	64.9	81.3	**87.3**	84.7	86.9
NFCorpus	32.2	26.0	34.2	**40.7**	35.8	36.2	37.8	38.2
NQ	30.6	46.7	56.8	–	48.2	52.4	62.4	**64.6**
HotpotQA	63.3	48.8	59.9	68.8	65.4	59.3	68.5	**70.1**
FiQA	23.6	25.2	46.7	**51.2**	41.1	37.2	45.8	48.5
ArguAna	39.7	26.5	54.0	43.5	**56.7**	51.4	48.6	56.5
Touche	**44.2**	25.9	25.6	29.1	28.0	25.4	30.5	34.2
Quora	78.9	78.7	**89.2**	63.8	87.6	84.6	86.8	88.3
DBPedia	31.8	31.4	40.8	43.2	40.2	39.9	43.7	**45.9**
SCIDOCS	14.1	11.3	16.1	–	18.6	**19.7**	18.1	18.9
FEVER	65.1	68.2	74.0	77.5	77.3	78.3	**83.4**	81.3
C-FEVER	16.5	18.7	26.7	22.3	23.7	30.5	31.0	**38.2**
SciFact	67.9	53.3	66.2	75.4	73.6	74.7	**75.6**	74.8
AVERAGE	43.7	40.1	49.3	–	52.1	52.1	55.1	**57.4**

Table 3: Zero-shot retrieval on BEIR benchmark. (The performances are measured by NDCG@10)

The above observations further affirm our previous conclusions about the advantage of LLM backbone and the benefit from LLaRA. It's worth noting that the LLM presents a powerful backbone to support document retrieval, because of not only its high expressiveness but also its long context, which enables the input document to be fully encoded instead of chunked into smaller segments.

4.3 Zero-shot Performance

We further investigate LLaRA's impact on the generalization. We leverage the BEIR benchmark for the evaluation of zero-shot performances, where the fine-tuned model from MS MARCO is directly applied to its included datasets (Table 3).

The major observations about the evaluation result are presented as follows. First of all, LLaRA exhibits a remarkable performance on BEIR, where it achieves a average performance of **57.4**. Such a performance is not only much higher than the rest of the baselines, but also establishes a new state-of-the-art result on BEIR (zero-shot). More impressively, it maintains the leading (bold) or the 2nd-place performance in almost every dataset, which indicates its superior versatility across different scenarios. Besides, LLaRA substantially outperforms RepLLaMA in most of the scenarios (11/13), which indicates the comprehensive improvement of the retriever's generalization. It is also worth to emphasize the comparison between BM25 and dense retrievers. When the BEIR benchmark was first launched two years ago, none of the dense retrievers (BERT and many others) were able to outperform BM25 despite their competitive performances in the supervised scenarios. However, the previous situation has been largely overturned with the adoption of LLM-based text encoders, as LLaRA outperforms BM25 on 12/13 datasets and goes beyond its average performance by 32% relatively. The dramatic improvement can attribute to three merits of LLMs: 1) the superior expressiveness to model complex semantics, 2) the expanded context to handle long inputs, and 3) the rich knowledge to understand common-sense relationships.

4.4 Technical Factors

Further study is made for three factors: unsupervised adaptation, prompt scheme, embedding size.

• **Adaptation**. Our previous experiments verify the effectiveness of unsupervised adaptation given its substantial improvement over LLaMA-2-7B. In this place, we focus on exploring the the underlying reason of the empirical advantage. As introduced, the unsupervised adaptation is performed to transform the text embedding such that it can become a global semantic representation and make accurate predictions for the vocabularies within the target context (§3.2). We perform the following experiment on MS MARCO to evaluate the adaptation effect. Firstly, the query and answer embeddings are projected into distributions in the vocabulary space: $\boldsymbol{d}_q \in \mathbb{R}^{|V|\times 1} \leftarrow \boldsymbol{e}_q^{\alpha T}\boldsymbol{W}, \boldsymbol{d}_a \in \mathbb{R}^{|V|\times 1} \leftarrow \boldsymbol{e}_a^{\beta T}\boldsymbol{W}$ ($\boldsymbol{W}$ is the decoding head of the LLM). Then, the top-N vocabs are predicted for the query and answer: $\boldsymbol{v}_q \leftarrow \textit{top-N}(\boldsymbol{d}_q), \boldsymbol{v}_a \leftarrow \textit{top-N}(\boldsymbol{d}_a)$. If the transformation works, there will be an increased lexical similarity between query and answer. In this place, we use BM25 to compute the similarity score: $\mathrm{BM}_{25}(\boldsymbol{v}_q, \boldsymbol{v}_a)$. As shown in Table 5, the adaptation (Adapt) exhibits a much higher vocab

Pro.	Avg	AR	CF	DB	FV	FQ	HQ	NF	NQ	QR	SD	SF	TO	TC
N2S	56.1	50.5	27.7	45.9	83.5	48.5	70.1	38.2	64.6	83.7	18.9	76.5	34.2	86.9
S2S	30.2	56.5	20.6	10.4	33.1	16.7	27.4	18.3	8.8	88.0	7.1	67.8	3.1	34.2
N2N	52.3	49.9	38.2	40.5	81.3	43.8	65.5	35.3	54.9	88.3	20.5	74.8	22.6	63.9
None	47.6	53.8	26.7	40.0	72.3	38.9	63.8	32.3	46.1	88.3	17.7	74.3	15.0	49.6
Ada*	57.4	56.5	38.2	45.9	81.3	48.5	70.1	38.2	64.6	88.3	18.9	74.8	34.2	86.9

Table 4: Impact of the adaptive usage of prompt (Ada*) evaluated on BEIR.

Top-N	Initial	Adapt	Fine-Tune
10	1.85	2.74	13.83
100	18.01	47.68	84.08
500	93.68	205.42	307.30
1000	219.09	392.80	542.89

Table 5: Impact on lexical similarity.

Dim.	DimRed	DimRed*	Sparse
768	41.0	41.2	41.5
1024	41.0	41.3	41.9
2048	41.1	41.4	42.3
4096	43.1	43.1	43.1

Table 6: Impact of embedding dimension.

similarity over the initial LLaMA-2 backbone (Initial). Interestingly, we also observe that the lexical similarity can be improved by fine-tuning as well. In fact, the improved lexical similarity is beneficial to the performance of dense retrieval, as studied by previous works on query and document expansion (Nogueira et al., 2019; Mao et al., 2020). With the unsupervised adaptation, such a transformation is implicitly accomplished for the text embedding.

• **Prompt**. The well-adapted LLM encoder is able to make adaptive usage of SELF and NEXT prompt to handle different correlation and paraphrasing relationships (Ada*). Particularly, NEXT-SELF (N2S) is used for correlation, SELF-SELF (S2S) is used for paraphrased documents, and NEXT-NEXT (N2N) is used for paraphrased short texts. The impact of prompt scheme is analyzed with BEIR, which contains retrieval tasks of diversified semantic relationships: for correlation, we have DBPedia (DB), FIQA (FQ), HotpotQA (HQ), NFCorpus (NF), NQ, SCIDOCS (SD), Touch (TO); for long-paraphrasing, we have Arguana (AR); for short paraphrasing, we have Climate-FEVER (CF), FEVER (FV), Quora (QR), SciFact (SF). The result is shown in Table 4, where the prompt utilization exerts a major impact. The majority of tasks are about correlation; thus, N2S works the most effectively in those scenarios. In other paraphrasing cases, S2S (e.g., Arguana) and N2N (e.g., Quora) can help to achieve a better result.

• **Dimension**. The LLMs are more expressive than smaller PLMs; however, they also come with higher costs in many perspectives. In this place, we focus on the impact from embedding size, i.e. dimension, which not only affects the computation but also determines the space cost of the vector database. We evaluate alternative dim reduction methods where the embedding size is gradually reduced from 4096 to 768 (Table 6). One is to jointly learn the LLM and a projection head during fine-tuning (DimRed); another one is to fixed the well fine-tuned LLM retriever and learn the projection head via distillation (Liu et al., 2022) (DimRed*). Unfortunately, both methods suffer from notably performance loss after dim reduction. We further replace linear projection with sparsification (Formal et al., 2021), where the top-N entries are selected for the embedding (Sparse). Compared with the first two options, sparsification turns out to be more effective in preserving retrieval performance. The above observations also suggest the necessity of light-weight processing of LLM-based retrievers in the future.

5 Conclusion

In this paper, we present LLaRA to enhance the LLM-based dense retrieval via unsupervised adaptation. LLaRA is composed of two pretext tasks, EBAE and EBAR, where the LLM is prompted to reconstruct the input sentence and predict the following sentence purely with its text embeddings. On top of the unsupervised adaptation, the LLM's text embedding can be transformed into a strong representation of global context. By using suitable prompts, it can flexibly support the semantic matching of different correlation and paraphrasing relationships. The effectiveness of LLaRA is verified by comprehensive experiments, where the adapted LLM achieves new state-of-the-art performances in both supervised and zero-shot evaluations, indicating the substantial improvements on both accuracy and generalization capability of the retrieval model.

6 Limitation

While LLaRA has made a substantial progress in adapting the LLM as a strong dense retriever, the current work can still be improved in the following ways. Firstly, the current method is only applied to a 7B model, it remain to explore its impact on larger LLMs. Secondly, the current model is for English centric scenario, it is necessary to make extensions for other languages. Thirdly, it is also important to find effective ways to maintain an efficient running cost for such large-scale embedding models.

7 Ethical Consideration

LLaRA is built upon LLaMA-2, it inherits potential biases, toxicity, and other problems present in the underlying LLM. Therefore, we do not recommend utilizing LLaRA for retrieval purposes in sensitive contexts. Moreover, the embedding may be influenced by the training data, potentially leading to biased or discriminatory results.

References

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

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

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,*, pages 4171–4186. Association for Computational Linguistics.

Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2021. Splade v2: Sparse lexical and expansion model for information retrieval. *arXiv preprint arXiv:2109.10086*.

Luyu Gao and Jamie Callan. 2021. Condenser: a pre-training architecture for dense retrieval. *arXiv preprint arXiv:2104.08253*.

Luyu Gao and Jamie Callan. 2022. Unsupervised corpus aware language model pre-training for dense passage retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2843–2853, Dublin, Ireland.

Luyu Gao, Zhuyun Dai, and Jamie Callan. 2021. Coil: Revisit exact lexical match in information retrieval with contextualized inverted list. *arXiv preprint arXiv:2104.07186*.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *SIGIR*, pages 113–122.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

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

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

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

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

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

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Zheng Liu and Yingxia Shao. 2022. Retromae: Pretraining retrieval-oriented transformers via masked auto-encoder. *arXiv preprint arXiv:2205.12035*.

Zheng Liu, Shitao Xiao, Yingxia Shao, and Zhao Cao. 2023. Retromae-2: Duplex masked auto-encoder for pre-training retrieval-oriented language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2635–2648.

Zhenghao Liu, Han Zhang, Chenyan Xiong, Zhiyuan Liu, Yu Gu, and Xiaohua Li. 2022. Dimension reduction for efficient dense retrieval via conditional autoencoder. *arXiv preprint arXiv:2205.03284*.

Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, and Xueqi Cheng. 2022. Pre-train a discriminative text encoder for dense retrieval via contrastive span prediction. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 848–858.

Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Xiang Ji, and Xueqi Cheng. 2021a. Prop: Pre-training with representative words prediction for ad-hoc retrieval. In *Proceedings of the 14th ACM international conference on web search and data mining*, pages 283–291.

Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Yingyan Li, and Xueqi Cheng. 2021b. B-prop: bootstrapped pre-training with representative words prediction for ad-hoc retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1513–1522.

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

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2020. Generation-augmented retrieval for open-domain question answering. *arXiv preprint arXiv:2009.08553*.

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

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al. 2022. Text and code embeddings by contrastive pretraining. *arXiv preprint arXiv:2201.10005*.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. *choice*, 2640:660.

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. 2021. Large dual encoders are generalizable retrievers. *arXiv preprint arXiv:2112.07899*.

Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to doctttttquery. *Online preprint*, 6:2.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2020. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2010.08191*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. Rocketqav2: A joint training method for dense passage retrieval and passage re-ranking. *arXiv preprint arXiv:2110.07367*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.

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

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022a. Simlm: Pre-training with representation bottleneck for dense passage retrieval. *arXiv preprint arXiv:2207.02578*.

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

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

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

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808*.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021. Optimizing dense retrieval model training with hard negatives. In *SIGIR*, pages 1503–1512.

Xin Zhang, Zehan Li, Yanzhao Zhang, Dingkun Long, Pengjun Xie, Meishan Zhang, and Min Zhang. 2023. Language models are universal embedders. *arXiv preprint arXiv:2310.08232*.

Kun Zhou, Yeyun Gong, Xiao Liu, Wayne Xin Zhao, Yelong Shen, Anlei Dong, Jingwen Lu, Rangan Majumder, Ji-Rong Wen, Nan Duan, et al. 2022. Simans: Simple ambiguous negatives sampling for dense text retrieval. *arXiv preprint arXiv:2210.11773*.