Unsupervised Text Representation Learning via Instruction-Tuning for Zero-Shot Dense Retrieval

Qiuhai Zeng[*†2], **Zimeng Qiu**[*1], **Dae Yon Hwang**[*1], **Xin He**[1], **William M. Campbell**[1]
[1]Amazon AGI
[2]Penn State University
qjz5084@psu.edu, {zimengqi dyhwang, xih, cmpw}@amazon.com

Abstract

Dense retrieval systems are commonly used for information retrieval (IR). They rely on learning text representations through an encoder and usually require supervised modeling via labelled data which can be costly to obtain or simply unavailable. In this study, we introduce a novel unsupervised text representation learning technique via instruction-tuning the pre-trained encoder-decoder large language model (LLM) under the dual-encoder retrieval framework. We demonstrate on multiple languages that the corpus representation can be augmented by the representations of relevant synthetic queries generated by the instruct-tuned LLM founded on the Rao-Blackwell theorem. Furthermore, we effectively align the query and corpus text representation with self-instruct tuning. We evaluate our proposed method under low-resource settings on three English, two German and one Portuguese retrieval datasets measuring NDCG@10, MRR@100, Recall@100. We significantly improve the average zero-shot retrieval performance on all metrics, increasing out-of-box FLAN-T5 model variations by $[4.73\%, 6.15\%]$ in absolute NDCG@10 and exceeding four supervised dense retrievers.

1 Introduction

Dense retrieval systems typically employ dual-encoder retrieval models which use two separate encoders, either symmetric or asymmetric, to represent the query and corpus in any languages (Gillick et al., 2018; Karpukhin et al., 2020b; Yang et al., 2020; Dong et al., 2022). The corpora are indexed with representation and will be retrieved in response to each query based on the relevance scores. The scores are usually calculated based on embedding similarity, such as dot product or cosine similarity. Although dense retrieval systems have developed rapidly, the model performance largely depends supervised text representation learning and relevancy capturing between the query and corpus (Zhao et al., 2022). Yet, it remains to be a major challenge to properly retrieve when lacking labeled modeling data. Existing work (Ni et al., 2022a,b) leveraged pre-trained large encoders (specifically T5 models, Raffel et al. (2020)) to alleviate the data thirst. However, their proposals still required annotated datasets either by web mining or manual annotation for fine-tuning in order to improve the generalization ability of dual-encoder retrieval models, for example, dealing with out-of-domain data. An alternative solution is to train a dense retrieval on synthetic query-corpus relevance pairs. Ma et al. (2021) trains a question generation system on general domain data and applies it to the targeted domain to construct synthetic question-passage data. To save the effort of training a task-specific generation model on general data, like Natural Questions (Kwiatkowski et al., 2019) or MSMARCO (Nguyen et al., 2016), Promptagator (Dai et al., 2023) proposes to use pre-trained large language models (LLMs), like FLAN (Wei et al., 2022), as a few-shot query generator to build the data for training the dual-encoder. However, the synthetic queries are not directly leveraged at inference, potentially causing gaps between training and inference of dense retrievers (Cho et al., 2022). Earlier work, e.g., doc2query (Nogueira et al., 2019b), concatenates the generated queries with the corresponding corpus, aiming to enrich the corpus representation with questions that the corpus can potentially answer. An improved version, docTTTTTquery (Nogueira et al., 2019a) leverages pre-trained T5 models as the expansion model, leading to more relevant synthetic queries and better retrieval performance.

Different from the previous work, we demonstrate directly on the embedding level instead of the text level, that the synthetically generated queries' embeddings can effectively augment the corpus rep-

[*]These authors contributed equally.
[†]Work done while intern at Amazon.

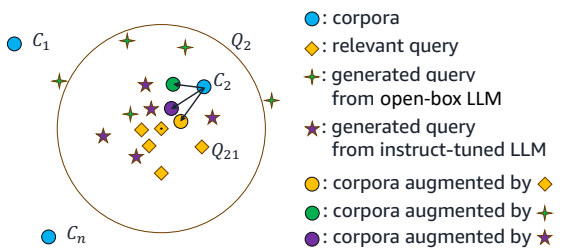

Figure 1: Illustration of the corpus representation augmented by embedding of relevant and synthetic queries generated by open-box and instruct-tuned LLMs.

resentation (Figure 1). Here, we propose an unsupervised representation learning approach through self-instructed-tuning leveraging the embedding generation and sequence generation capability of an encoder-decoder LLM. This approach consists of two steps, i.e., self-instructed-learning and Rao-Blackwellization. In the first step, we design two instruction tasks, namely question generation and keyword summarization, to generate synthetic questions and keywords for each given corpus via prompting a pre-trained LLM. Next, we apply filters to gate the synthetic data quality and instruction-tune the pre-trained LLM (and its variant versions) on the filtered output (Step one in Figure 2). In the second step, we use the instruct-tuned LLM to generate better synthetic questions and keywords following the same instruction prompts as in training. We then obtain the embeddings of the newly generated synthetic questions and keywords and that of corpus from the instruct-tuned encoder, and take the weighted average as our augmented corpus representation (Step two in Figure 2).

We consider the corpus representation learning task as a problem of query embedding expectation estimation. Based on the Rao-Blackwell theorem, the crude estimator, corpus embedding, can be improved by taking the conditional expectation given the sufficient statistics, i.e., sample mean of the embedding of their (synthetic) relevant queries and keywords. Thus, we expect combining the raw corpus embedding and synthetic query embedding to achieve better corpus representation. Besides, by aligning instruction-tuning and synthetic query generation, the retrieval model is directly optimized on corpus representation during training. To assess the effectiveness of our proposed method, we compare retrieval method of corpus only embedding with our augmented corpus representation, models with and without instruction-tuning and evaluate against four competitive dense retrievers (i.e., mDPR (Zhang et al., 2021, 2022), mBART (Tang et al., 2020), T-Systems (T-Systems, 2020), Albertina-PT (Santos et al., 2024)). Our main contributions are as follows:

- We propose a novel unsupervised text representation learning approach for information retrieval (IR) by instruction-tuning a pre-trained encoder-decoder with unlabelled corpus.
- We demonstrate our approach of using conditional expectation of the relevant (synthetic) query/keywords embedding the representation of the corpus can be augmented effectively, founded on the Rao-Blackwell theorem.
- We verify the effectiveness of the proposed methods on three English, two German and one Portuguese IR datasets. We significantly improve the zero-shot average retrieval performance with our unsupervised approach and exceed four competitive supervised dense retrievers (Table 5 - 7).

2 Related Work

2.1 Instruction-tuning

Tuning pre-trained LLMs with *(natural language instruction, response)* pairs to enhance models' ability to follow instructions and understand user intention. It is a rising paradigm in natural language processing (NLP) to strengthen model's generalizability on unseen tasks. FLAN (Wei et al., 2022) significantly improves a 137B LLM's zero-shot performance via instruction learning on various NLP datasets with multiple instruction templates. InstructDial (Gupta et al., 2022) also shows significant zero-shot performance boost in unseen dialogues when applying instruction-tuning to dialogue domain. InstructGPT (Ouyang et al., 2022) enhances GPT-3's performance by fine-tuning it on instructions and human feedback collected from OpenAI API. Self-Instruct (Wang et al., 2023) fine-tunes the open-box GPT-3 on its own generated instructions and instances which achieved on par performance of InstructGPT.

2.2 Dense Retrieval Text Representation

Text representation is the foundational component of dense retrieval. Under dual-encoder framework, it has been a long standing practice to represent query and corpus with encoder only models, e.g.,

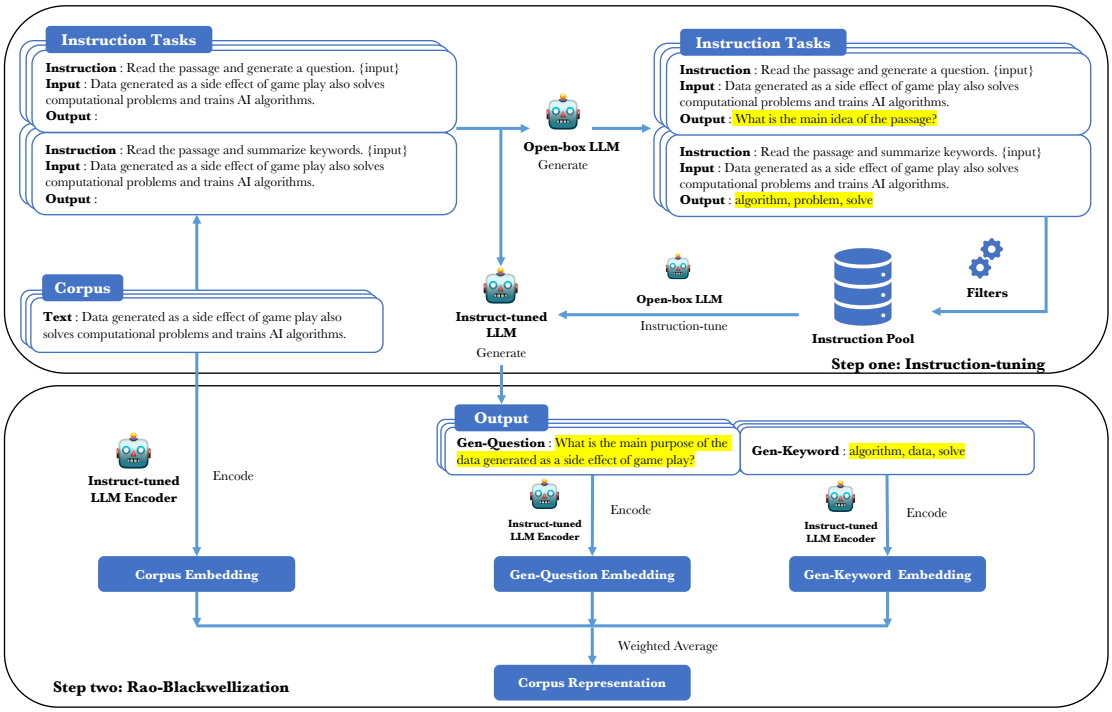

Figure 2: A high-level overview of Encoder-Decoder corpus representation with our approach. In the instruction-tuning step, given a set of instruction tasks (in our case **keyword summarization**: "Read the passage and summarize keywords." and **question generation**: "Read the passage and generate a question."), the pre-trained LLM will generate instruction following examples which are passed through filters for quality control. The filtered examples form an instruction pool and are used to instruction-tune the open-box LLM. In the Rao-Blackwellization step, by prompting the instruct-tuned LLM using the same instructions as before, synthetic questions and keywords are generated for the corpus. Both the corpus and the generated sequences are encoded by the LLM encoder and the weighted average of their embedding is used as corpus representation.

BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), such as in Sentence-BERT (Reimers and Gurevych, 2019), ColBERT (Khattab and Zaharia, 2020). Recently, Sentence-T5 (Ni et al., 2022a) demonstrates superior performance with encoder-decoder pre-trained LLM, T5. Moreover, representing corpus with single representation may not well model the fine-grained semantic interaction between the queries and corpus. Poly-encoder (Humeau et al., 2019) and ME-BERT (Luan et al., 2020) learn multiple representations to better capture the corpus semantics and show significant improvement. Doc2query (Nogueira et al., 2019b) and docTTTTTquery (Nogueira et al., 2019a) append generated synthetic queries to the corpus and thus enrich the semantic information.

3 Method

We propose an unsupervised text representation learning approach through self-instructed-tuning a pre-trained encoder-decoder LLM. First, we generate instruction following responses from an LLM and instruction-tune the LLM itself with filtered quality *(natural language instruction, response)* pairs. Next, we compute the augmented corpus embedding weighing in synthetic queries' (e.g., questions, keywords) embeddings. Figure 2 presents the overall flow of our approach.

3.1 Problem Scenario

Denote corpora as C_1,C_2,...,C_n, and their relevant queries as Q_{11}, Q_{12},...,Q_{21},..., where queries Q_{i1},Q_{i2},... are relevant to the same corpora C_i. For example, Q_{11} can be `Harry Potter 1` and Q_{12} can be `Harry Potter and the Philosopher's Stone`, whereas C_1 is `Harry Potter and the Sorcerer's Stone`. $Q_i = Q_{i1}, ..., Q_{im}$

Given a pre-trained encoder-decoder LLM, besides treating the encoder as a text representation model, we consider it as a random variable, where the sample space consists of the range of the possible embedding values, and the corresponding probability measure to each text portion.

$$\texttt{Encoder}(\cdot) : text \mapsto embedding \quad (1)$$

where the embedding refers to the sentence embedding of the text.

We assume that an effective encoder maps each group of Q_i near a group center in the high-dimensional space and also maps the corresponding C_i to the surrounding area so that Q_i and C_i are well associated. For example, given $Q_{21} \in Q_2$ query, the retrieval system will retrieve the C_2 corpora which is the closest to the query (Figure 1).

Corpus Embedding as an Expectation Estimator The group center is a comprehensive depiction of the entire group and is indicative to distinguish from other groups. With the pre-trained $\texttt{Encoder}(\cdot)$, the group center is essentially the expected value of each group queries' embeddings, denoted by $\mathbb{E}(\texttt{Encoder}(Q_i))$. When we use the embedding of the corpus, i.e., $\texttt{Encoder}(C_i)$, as its representation, we are using it to estimate the group center $\mathbb{E}(\texttt{Encoder}(Q_i))$. This is effective when we don't have any information from the query group.

Application of the Rao–Blackwell theorem Assume we have relevant queries Q_{i1},Q_{i2}, ..., Q_{im} for corpus C_i. Then $\frac{1}{m}\sum_{j=1}^{m}$ $\texttt{Encoder}(Q_{ij})$ is a sufficient statistics to estimate $\mathbb{E}(\texttt{Encoder}(Q_i))$.

According to Rao–Blackwell Theorem: If $g(\mathbf{X})$ is any kind of estimator of a parameter θ, then the conditional expectation of $g(\mathbf{X})$ given $T(\mathbf{X})$, namely $\mathbb{E}(g(x)|T(x))$, where T is a sufficient statistic, is typically a better estimator of θ, and is never worse. Plug in Equation (2), we get an improved estimator for $\mathbb{E}(\texttt{Encoder}(Q_i))$, which is $\mathbb{E}(\texttt{Encoder}(C_i)|\frac{1}{m}\sum_{j=1}^{m}\texttt{Encoder}(Q_{ij}))$.

$$\begin{aligned} g(x) &= \texttt{Encoder}(C_i) \\ T(x) &= \frac{1}{m}\sum_{j=1}^{m}\texttt{Encoder}(Q_{ij}) \\ \theta &= \mathbb{E}(\texttt{Encoder}(Q_i)) \end{aligned} \tag{2}$$

With some regularity assumptions, e.g., $C_i \in Q_i$ and $C_i = Q_{i1}$, the conditional expectation can be written as

$$\begin{aligned} &\mathbb{E}(\texttt{Encoder}(C_i)|\frac{1}{m}\sum_{j=1}^{m}\texttt{Encoder}(Q_{ij})) \\ &= \frac{1}{m}\sum_{j=1}^{m}\texttt{Encoder}(Q_{ij}) \\ &= \frac{1}{m}\texttt{Encoder}(C_i) + \frac{1}{m}\sum_{j=2}^{m}\texttt{Encoder}(Q_{ij}) \end{aligned} \tag{3}$$

We expect to achieve better performance with this formula for corpus representation. An intuitive understanding is that it gets closer to the relevant queries' embedding in the vector space (Figure 1).

3.2 Synthetic Query Generation

Obtaining a comprehensive set of labeled queries is labor-intensive and costly, especially in low resource setting. LLMs are known for its generative capability following well designed instructions. Not only can the model generate text, but it also can output the generation probability of the text. We denote the generation model by $\texttt{LLM}(\cdot)$, then the generation can be written as

$$\hat{Q}_{ij}, \hat{P}(\hat{Q}_{ij}) = \texttt{LLM}(\texttt{Instruction} + C_i) \tag{4}$$

where $\hat{Q}_{ij}$ is the generated query and $\hat{P}(\hat{Q}_{ij})$ is the generation probability. The instruction is a pre-defined generation task, for example "write a question for" or "what are the keywords of".

3.3 Corpus Representation

Plug in the synthetic queries, let $R(C_i)$ denote the final representation of corpora C_i, Equation (3) becomes a weighted average of the original corpora embedding and its synthetic query embedding,

$$\begin{aligned} R(C_i) \hat{=}\ & w_0\texttt{Encoder}(C_i) + \\ & (1-w_0)\sum_j \hat{P}(\hat{Q}_{ij})\texttt{Encoder}(\hat{Q}_{ij}) \end{aligned} \tag{5}$$

where w_0 is a hyper-parameter that is tuned on a subset of test queries. Equation (5) is our proposed corpus representation for the dual-encoder retrieval system. Note that we can generate different types of synthetic queries in Equation (4) using various instructions, and we can generate multiple sequences for each instruction by adopting decoding strategies such as beam search. We can also improve the quality of the generated queries through instruction-tuning as follows.

3.4 Instruction-Tuning the LLM

While LLM has reasonable text generation capabilities, its ability to precisely follow specific instructions can be honed via instruction-tuning.

As we don't have the query-corpora labeled data, we propose to self-instructed-tuning the LLM on its self-generated quality (i.e., gated) responses following given instructions to enhance synthetic queries generation relevance. This approach has demonstrated its effectiveness (Wang et al., 2023). The instruct-tuned LLM is then used to prepare the

synthetic queries for the corpus representation augmentation as in Equation (6).

$$\hat{Q}_{ij}, \hat{P}(\hat{Q}_{ij}) = \texttt{InstructTunedLLM}(\texttt{Instruction} + C_i) \quad (6)$$

We use the same instructions across the entire framework, including generation and training. Figure 1 shows a schematic diagram that although the generated queries from an open-box pre-trained LLM may not effectively enrich the corpora, after instruction-tuning, the generated synthetic queries become more relevant and the corpus representation can be improved consequently.

4 Experiments

4.1 Datasets

In this work, we tested six IR datasets where the summary of the database is shown in Table 1. ***English:*** (1) NFCorpus (Boteva et al., 2016) has automatically extracted relevance judgments for medical documents. (2) SciFact (Wadden et al., 2020) consists of expert-annotated scientific claims with abstracts and rationales. (3) SCIDOCS (Cohan et al., 2020) has seven document-level tasks from citation prediction, document classification, and recommendation. ***German:*** (4) GermanQuAD (Möller et al., 2021) has the relevant information for high complex German QA with a large size of corpora. (5) GermanDPR (Möller et al., 2021) is a passage retrieval dataset which shares the same corpus as GermanQuAD. ***Portuguese:*** (6) mMARCO/PT (Bonifacio et al., 2021) is translated version of MS MARCO (Bajaj et al., 2018) in Portuguese with anonymized questions from Bing's search query logs. Due to computation resource limits, we downsample the corpus in SCIDOCS, GermanQuAD, GermanDPR and mMARCO/PT datasets, where we ensure the downsampled corpus include all relevant corpus for test queries. Note that such downsampling does not prevent us from fairly comparing the zero-shot retrieval efficacy of our approach with open-box LLMs because all experiments are performed under the same data setting. To help the encoder capture the fine-grained semantic interaction between queries and corpus, we divide each corpora into multiple sentences using the PunktSentenceTokenizer [1] from nltk package and use the sentence level multi-representation for the corpora, meaning that if any of the sentence is retrieved, the passage is retrieved.

[1] https://www.nltk.org/api/nltk.tokenize.PunktSentenceTokenizer.html

Table 1: Details of datasets used. The size of corpus is downsampled to 15K in SCIDOCS, 10K in GermanQuAD and GermanDPR, and 7K in mMARCO/PT. Filtered Queries: Generated synthetic queries from FLAN-T5-Large with filtering.

Dataset	Language	#Test Queries	Corpus Size	#Filtered Queries
NFCorpus	English	323	3.6K	5.9K
SciFact	English	300	5.1K	8.2K
SCIDOCS	English	1K	25.6K	29.4K
GermanQuAD	German	2K	2.8M	17.5K
GermanDPR	German	1K	2.8M	17.5K
mMARCO/PT	Portuguese	6K	8.8M	12.7K

Table 2: Average performance of FLAN-T5 with out-of-box encoder-only embedder on Passage vs Sentence level indexing. Metrics: ♠: NDCG@10, ♣: MRR@100, ♡: Recall@100.

Models	♠	♣	♡
Base (Passage)	8.1	9.1	29.8
Large (Passage)	12.0	12.6	41.1
Base (Sentence)	23.1	25.0	49.0
Large (Sentence)	**24.9**	**26.4**	**52.1**

4.2 Baseline

We compare between the corpus-only representation and our proposed augmented corpus representation in zero-shot experiments under the dual-encoder framework. To obtain the representation of a sequence from the encoder, we perform mean aggregation over the last hidden state of each token (Ni et al., 2022a). We measure the relevance between query and corpus using cosine similarity.

To understand the superiority of our approach, we compare with four different state-of-the-art (SOTA) models: (1) mDPR (Zhang et al., 2021, 2022) is a variation of DPR model (Karpukhin et al., 2020a) which replaces BERT to multilingual BERT (Devlin et al., 2019) to support non-English languages for retrieval tasks. (2) mBART-Large (Tang et al., 2020) is a multilingual Sequence-to-Sequence generation model. It supports 50 languages and we consider it for comparison in same model structure (i.e., encoder-decoder). (3) T-Systems (T-Systems, 2020) is developed for computing sentence embeddings for English and German texts. It uses a XLM-RoBERTa (Conneau et al., 2019) and is fine-tuned with English-German datasets. (4) Albertina-PT (Santos et al., 2024) is a

Table 3: Comparison of model performances with and without instruction-tuning. Base/Large: out-of-box FLAN-T5-Base/Large. Instruct-Base/Large: FLAN-T5-Base/Large with instruction-tuning. Metrics: ♠: NDCG@10, ♣: MRR@100, ♡: Recall@100.

Models	NFCorpus			SciFact			SCIDOCS			GermanQuAD			GermanDPR			mMARCO/PT			**Average**		
	♠	♣	♡	♠	♣	♡	♠	♣	♡	♠	♣	♡	♠	♣	♡	♠	♣	♡	♠	♣	♡
Base	12.2	26.6	15.8	29.6	28.5	66.3	6.4	13.4	17.7	49.4	45.8	83.2	41.5	37.8	81.3	20.4	19.1	51.0	26.6	28.5	52.6
Large	10.4	23.4	14.6	30.7	28.8	71.5	**7.2**	14.1	22.1	50.8	47.2	83.6	47.4	43.5	82.2	25.0	23.3	**57.06**	28.6	30.1	55.2
Instruct-Base	**12.3**	27.0	**16.2**	30.7	29.6	65.1	6.0	12.7	16.5	52.4	48.5	84.4	42.6	38.5	81.4	28.8	31.3	52.7	27.8	29.7	53.1
Instruct-Large	11.9	**27.0**	15.9	**32.0**	**29.9**	**73.2**	7.2	**14.6**	**22.4**	**55.5**	**52.0**	**86.8**	**51.6**	**48.1**	**85.5**	**31.6**	**34.3**	56.7	**31.1**	**33.0**	**57.5**

Table 4: Example of synthetic queries from FLAN-T5-Large according to the instruction-tuning.

Corpus	**Instruction Type**	**Open-box**	**Instruct-tuned**
Fluorometric titration of E. coli single-stranded DNA binding protein with various RNAs showed that the protein specifically and cooperatively binds to its own mRNA. The binding inhibited in vitro expression of ssb and bla but not nusA. This inhibition takes place at a physiological concentration of SSB. The function of the protein in gene regulation is discussed.	Keyword	The single-stranded DNA binding protein(SSB) specifically and cooperative -ly binds to its own mRNA.	mRNA, protein, titration
This paper describes an aggregation and correlation algorithm used in the design and implementation of an intrusion-detection console built on top of the Tivoli Enterprise Console (TEC). The aggregation and correlation algorithm aims at acquiring intrusion-detection alerts and relating them together to expose a more condensed view of the security issues raised by intrusion-detection systems.	Question	What is the purpose of the paper?	What is the purpose of the aggregation and correlation algorithm?

foundational model for European Portuguese. It is based on the DeBERTa model (He et al., 2021) and we consider this model as the competitive model in Portuguese dataset. Lastly, we compare with docTTTTTquery (Nogueira et al., 2019a) to understand the effectiveness of our corpus representation augmentation.

4.3 Encoder-Decoder Models

T5 is an encoder-decoder model pre-trained on a combination of unsupervised and supervised tasks, where each task is transformed into a text-to-text format (Raffel et al., 2020). FLAN-T5 is an enhanced version of T5 fine-tuned on a mixture of tasks (Wei et al., 2022). Considering that these types of models are open source, offer various sizes, support English, German and Portuguese, and have an encoder-decoder architecture, we leverage the FLAN-T5-Base and Large models in our experiments.

4.4 Instruction Query Generation

For instruction query generation and instruction-tuning, we consider two types of instructions (i.e., keyword summarization and question generation) as shown in Figure 2. We also develop a filter to improve the quality of generated instructions. If the task is keyword summarization, the number of keywords should be smaller than the half number of sentences in corpus. If it's question generation, the generated sequence should end with a question mark. The filter is simple, leaving room for further improvement. The numbers of the filtered synthetic queries are shown in Table 1.

4.5 Hyperparameter Setting

When performing instruction-tuning, we use the same hyperparameter setting for all the models. Specifically, we use the AdaFactor optimizer with learning rate 0.0001, batch size 16, and the number of epochs 30. Early stopping is performed when the validation loss shows no improvement for five consecutive epochs.

When generating queries using FLAN-T5 models, we only consider one returned sequence for each instruction and assume they are equally important. We denote the generated question and keywords as $\hat{question}_i$ and $\hat{keywords}_i$. We tested the multiple weighting methods for corpus representation where the best approach is giving the weight on the original corpus as $w_0 = 0.6$, so that each of $\hat{question}_i$ and $\hat{keywords}_i$ has the weight 0.2. Thus, the corpus representation is:

$$\mathsf{R}(C_i) = 0.6 \times \mathsf{Encoder}(C_i) + 0.2 \times (\mathsf{Encoder}(\hat{question}_i) + \mathsf{Encoder}(\hat{keywords}_i)) \quad (7)$$

5 Results and Discussion

5.1 Corpora vs Sentence Indexing

We evaluate whether the sentence level multi-representation can capture the semantic interaction between the corpora and the query. Results for FLAN-T5 models using encoder-only representation are shown in Table 2. The sentence level multi-representation embedding technique outperforms the corpora level single representation by a large margin across all datasets. As the model size increases, the performance also gets better. Note that our approach uses no labeled data to achieve on par performance as SOTA models, and sentence level indexing is a way we do for chunking. According to the promising empirical results, we will apply the sentence level multi-representation technique to all the following experiments.

5.2 Overall Results

Table 3 describes the performance of FLAN-T5 models regarding instruction-tuning. Overall, we can mostly find the improvements of performances in all metrics after instruction-tuning, especially in non-English. This is mainly because the quality of generated queries after instruction-tuning are proper and detailed (Table 4), and also each synthetic query is less overlapped which makes the corpora distinguishable. The influence of instruction-tuning is mostly greater in larger model since it can have better generation capability and be more affected by fine-tuning with instructions.

Table 5 - 7 compare ours with SOTA models in zero-shot scenarios. In English datasets (Table 5), instruct-tuned FLAN-T5-Base mostly outperforms other baselines, except for T-Systems which is enhanced model for English and German and has a bigger size. With instruct-tuned FLAN-T5-Large, we exceeds all others in terms of average performances. In German datasets (Table 6), instruct-tuned FLAN-T5-Base shows the better overall performances with smaller size which emphasizes the resource-effectiveness of our approach. When we consider the larger model, we significantly outperforms other SOTAs. Lastly, in Portuguese dataset (Table 7), we slightly underperform than the competitive baseline which only supports the single language. By considering the larger model with instruct-tuning, we exceed others with large gap. Overall, our approach shows the effectiveness in all languages, especially in non-English datasets.

5.3 Ablation Study

To deeply understand the effectiveness of our approach, we did the solid ablation study where we exclude the GermanDPR and mMARCO/PT for this study which always shows the similar pattern.

Optimal Corpus Representation From our findings, new corpus representation based on synthetic queries from instructions is useful to improve retrieval performances. To define the optimal weights in corpus representation, we investigate four different weighting methods: (1) Equal: giving equal weights for corpus and synthetic queries (i.e., keyword, question). (2) Manual: same as Equation (7). (3) `BERTScore`: Assigning the weights based on `BERTScore` (F1) with BERT (Multilingual-Cased) model (Devlin et al., 2018) as shown in Equation (8), where X denotes $\hat{keywords}_i, \hat{question}_i$. (4) $\texttt{BERTScore}_{Softmax}$: applying Softmax on top of `BERTScore`.

$$\begin{aligned} \texttt{Weight}_X &= \frac{\texttt{BERT}(X, C_i)}{1 + \texttt{Sum}(\texttt{BERT}(X, C_i)}, \\ \texttt{Weight}_{C_i} &= \frac{1}{1 + \texttt{Sum}(\texttt{BERT}(X, C_i))} \end{aligned} \tag{8}$$

Table 8 shows the overall performances of different weight approaches in corpus representation. Firstly, the equal weight approach shows the worst performance which confirms that the corpus basically contains the most relevant information for queries which should be weighted more. Also, extracted keywords and questions mostly have the essential contexts but partial information of corpus which is not enough to include the semantic meaning of corpus. Thus, manual weighting with emphasis on corpus promises better result than BERTScore approaches. Lastly, we generated the corpus representation based on text-level concatenation (Nogueira et al., 2019a) where we confirm the superiority of embedding-level representations.

Effectiveness of Instruction-tuning Table 4 gives the examples of generated synthetic queries. In keyword summarization, open-box extracts a simple copy of sentence as keywords while instruction-tuning helps to observe the whole corpus to extract the core keywords. For question generation, open-box generates the general question while instruction-tuning gives the detailed and suitable questions which can be accountable by the specific corpus.

Figure 3 shows the distributions of embeddings of corpora and test queries with FLAN-T5-Large. Overall, the weighted corpus representation and

Table 5: Comparison with SOTA models (size) on English datasets. Instruct-Base/Large: FLAN-T5-Base/Large with instruction-tuning. Metrics: ♠: NDCG@10, ♣: MRR@100, ♡: Recall@100.

Models	NFCorpus			SciFact			SCIDOCS			**Average**		
	♠	♣	♡	♠	♣	♡	♠	♣	♡	♠	♣	♡
mDPR (177M)	8.3	19.2	11.6	23.5	21.9	58.9	4.8	10.3	16.0	12.2	17.1	28.8
T-Systems (278M)	**15.3**	**29.1**	**17.1**	25.3	23.7	59.3	**8.4**	**17.6**	**23.8**	16.3	**23.5**	**33.4**
mBART-Large (331M)	1.9	5.9	4.6	23.9	22.5	52.5	3.6	7.8	12.7	9.8	12.1	23.3
Instruct-Base (109M)	12.3	27.0	16.2	**30.7**	**29.6**	**65.1**	6.0	12.7	16.5	**16.4**	23.1	32.6
Instruct-Large (341M)	11.9	27.0	15.9	***32.0***	***29.9***	***73.2***	7.2	14.6	22.4	***17.0***	***23.8***	***37.2***

Table 6: Comparison with SOTA models (size) on German datasets. Instruct-Base/Large: FLAN-T5-Base/Large with instruction-tuning. Metrics: ♠: NDCG@10, ♣: MRR@100, ♡: Recall@100.

Models	GermanQuAD			GermanDPR			**Average**		
	♠	♣	♡	♠	♣	♡	♠	♣	♡
T-Systems (278M)	33.9	31.0	64.1	**53.4**	**49.6**	**83.5**	43.7	40.3	73.8
mBART-Large (331M)	34.1	31.5	63.3	30.8	27.4	64.2	32.5	29.5	63.8
Instruct-Base (109M)	**52.4**	**48.5**	**84.4**	42.6	38.5	81.4	**47.5**	**43.5**	**82.9**
Instruct-Large (341M)	***55.5***	***52.0***	***86.8***	51.6	48.1	***85.5***	***53.5***	***50.1***	***86.1***

Table 7: Comparison with SOTA on Portuguese dataset **mMARCO/PT**. Instruct-Base: FLAN-T5-Base with instruction-tuning. Metrics: ♠: NDCG@10, ♣: MRR@100, ♡: Recall@100.

Metric	Albertina-PT (139M)	mBART-Large (331M)	Instruct-Base (109M)
♠	**23.7**	2.3	22.9
♣	**22.0**	2.2	21.6
♡	**57.1**	18.3	55.1

instruction-tuning spread out the corpora embeddings to make them distinguishable. It also helps to locate the test queries closer to the corpora. Thus, our approach helps to integrate the crucial and detailed synthetic queries for corpus representation that leads to unique corpora representation to achieve enhanced retrieval performances.

Effectiveness of Corpus Representation Augmentation We compare with other corpus representation augmentation, docTTTTTquery (Nogueira et al., 2019a), to validate our corpus augmentation. Here, we follow the default strategy of docTTTTTquery: top-10 with 40 predictions appending on corpus. According to Table 9, we demonstrate significant improvement via our approach - embedding level augmentation with representations from self-instructed-tuned model. Based on this finding, we can confirm that augmenting representation on embedding level is more effective than on input text level with concatenation as docTTTTTquery, and our self-instructed-tuned model performs better than their supervised representation generation model.

Table 8: Effects of different weight methods for corpus representation with FLAN-T5. Concatenation means the appending corpus with synthetic queries in text-level while others are done in embedding-level. Metrics: ♠: NDCG@10, ♣: MRR@100, ♡: Recall@100.

Corpus Weights	**Models**	♠	♣	♡
N/A	Base	22.0	26.0	43.5
	Large	23.2	26.5	46.2
Equal	Base	18.3	22.0	38.8
	Large	17.9	21.6	39.9
Manual	Base	24.4	**28.6**	45.8
	Large	**24.8**	28.4	**47.9**
BERTScore	Base	22.4	26.1	43.6
	Large	22.0	25.5	45.2
BERTScore$_{Softmax}$	Base	20.1	23.6	40.7
	Large	19.5	23.1	42.7
Concatenation	Base	15.8	18.9	36.7
	Large	15.6	19.1	36.9

Table 9: Effects of different corpus representation augmentation with FLAN-T5. Metrics: ♠: NDCG@10, ♣: MRR@100, ♡: Recall@100.

Models	♠	♣	♡
docTTTTTquery (Base)	9.6	12.8	24.9
Our approach (Base)	**22.0**	**26.0**	**43.5**
docTTTTTquery (Large)	13.4	16.3	33.3
Our approach (Large)	**23.2**	**26.5**	**46.2**

6 Conclusion

In our research, we propose the unsupervised text representation learning technique through self-

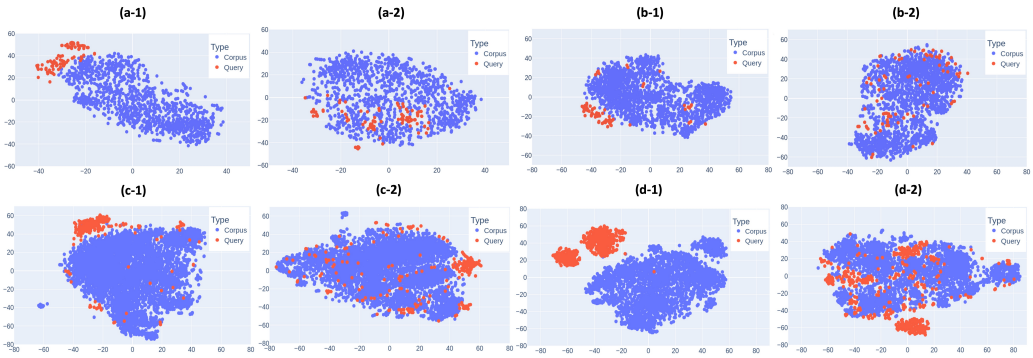

Figure 3: t-SNE distributions for corpus representation generated from FLAN-T5-Large. (a-d) NFCorpus, SciFact, SCIDOCS, GermanQuAD. (1-2) Original corpus, Weighted corpus with synthetic queries after instruction-tuning.

instructed-tuning encoder-decoder LLMs. Based on the Rao-Blackwell theorem, we leverage the embeddings of synthetically generated queries (i.e., questions and keywords) to augment the corpus representation for the dual-encoder retrieval framework. In zero-shot experiments, our proposed corpus representation consistently improves the performance over encoder-only corpus representation. Even if the open-box LLM was not pre-trained on retrieval task and there is no labeled modeling data, after fine-tuning with our approach it exceeds the SOTA models across different datasets, presenting the high effectiveness and data efficiency of our method in retrieval tasks.

In future work, we plan to explore our proposed method on separate encoder and decoder models.

References

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. 2018. Ms marco: A human generated machine reading comprehension dataset.

Luiz Henrique Bonifacio, Vitor Jeronymo, Hugo Queiroz Abonizio, Israel Campiotti, Marzieh Fadaee, , Roberto Lotufo, and Rodrigo Nogueira. 2021. mmarco: A multilingual version of ms marco passage ranking dataset.

Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. 2016. A full-text learning to rank dataset for medical information retrieval.

Sukmin Cho, Soyeong Jeong, Wonsuk Yang, and Jong C. Park. 2022. Query generation with external knowledge for dense retrieval. In *Proceedings of Deep Learning Inside Out: The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures, DeeLIO@ACL 2022, Dublin, Ireland and Online, May 27, 2022*, pages 22–32. Association for Computational Linguistics.

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

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Zhuyun Dai, Vincent Y. Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B. Hall, and Ming-Wei Chang. 2023. Promptagator: Few-shot dense retrieval from 8 examples. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805.

Zhe Dong, Jianmo Ni, Dan Bikel, Enrique Alfonseca, Yuan Wang, Chen Qu, and Imed Zitouni. 2022. Exploring dual encoder architectures for question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 9414–9419. Association for Computational Linguistics.

Daniel Gillick, Alessandro Presta, and Gaurav Singh Tomar. 2018. End-to-end retrieval in continuous space. *ArXiv*, abs/1811.08008.

Prakhar Gupta, Cathy Jiao, Yi-Ting Yeh, Shikib Mehri, Maxine Eskénazi, and Jeffrey P. Bigham. 2022. Instructdial: Improving zero and few-shot generalization in dialogue through instruction tuning. In *Conference on Empirical Methods in Natural Language Processing*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *International Conference on Learning Representations*.

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

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020b. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics.

O. Khattab and Matei A. Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

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

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. *ArXiv*, abs/1907.11692.

Y Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2020. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345.

Ji Ma, Ivan Korotkov, Yinfei Yang, Keith B. Hall, and Ryan T. McDonald. 2021. Zero-shot neural passage retrieval via domain-targeted synthetic question generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 1075–1088. Association for Computational Linguistics.

Timo Möller, Julian Risch, and Malte Pietsch. 2021. Germanquad and germandpr: Improving non-english question answering and passage retrieval. *ArXiv*, abs/2104.12741.

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. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Jianmo Ni, Gustavo Hernández Ábrego, Noah Constant, Ji Ma, Keith B. Hall, Daniel Cer, and Yinfei Yang. 2022a. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1864–1874. Association for Computational Linguistics.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y. Zhao, Yi Luan, Keith B. Hall, Ming-Wei Chang, and Yinfei Yang. 2022b. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 9844–9855. Association for Computational Linguistics.

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

Rodrigo Frassetto Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019b. Document expansion by query prediction. *ArXiv*, abs/1904.08375.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Conference on Empirical Methods in Natural Language Processing*.

Rodrigo Santos, João Rodrigues, Luís Gomes, João Silva, António Branco, Henrique Lopes Cardoso, Tomás Freitas Osório, and Bernardo Leite. 2024. Fostering the ecosystem of open neural encoders for portuguese with albertina pt-* family.

T-Systems. 2020. T-system model.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning.

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

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernández Ábrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 87–94. Association for Computational Linguistics.

Xinyu Zhang, Xueguang Ma, Peng Shi, and Jimmy Lin. 2021. Mr. TyDi: A multi-lingual benchmark for dense retrieval. *arXiv:2108.08787*.

Xinyu Zhang, Kelechi Ogueji, Xueguang Ma, and Jimmy Lin. 2022. Towards best practices for training multilingual dense retrieval models.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 2022. Dense text retrieval based on pretrained language models: A survey. *ArXiv*, abs/2211.14876.