What Are You Token About? Dense Retrieval as Distributions Over the Vocabulary

Ori Ram[1] **Liat Bezalel**[1] **Adi Zicher**[1]
Yonatan Belinkov[2] **Jonathan Berant**[1] **Amir Globerson**[1]
[1]Blavatnik School of Computer Science, Tel Aviv University [2]Technion – IIT, Israel
`ori.ram@cs.tau.ac.il, liatbezalel@mail.tau.ac.il, adizicher@mail.tau.ac.il`
`belinkov@technion.ac.il, joberant@cs.tau.ac.il, gamir@tauex.tau.ac.il`

Abstract

Dual encoders are now the dominant architecture for dense retrieval. Yet, we have little understanding of how they represent text, and why this leads to good performance. In this work, we shed light on this question via *distributions over the vocabulary*. We propose to interpret the vector representations produced by dual encoders by projecting them into the model's vocabulary space. We show that the resulting distributions over vocabulary tokens are intuitive and contain rich semantic information. We find that this view can explain some of the failure cases of dense retrievers. For example, the inability of models to handle tail entities can be explained via a tendency of the token distributions to *forget* some of the tokens of those entities. We leverage this insight and propose a simple way to *enrich* query and passage representations with lexical information at *inference* time, and show that this significantly improves performance compared to the original model in out-of-domain settings.

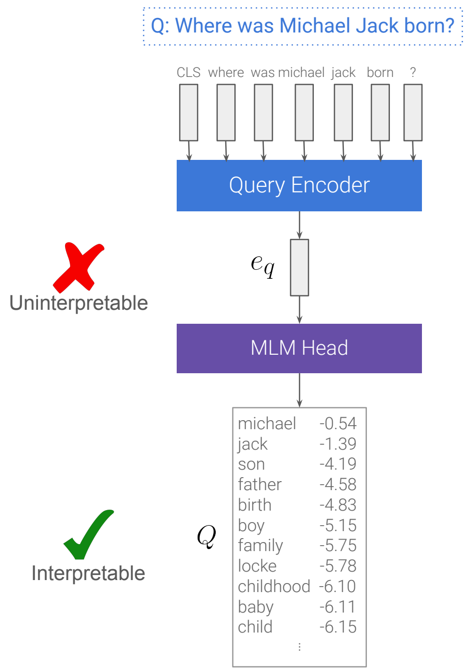


Figure 1: An example of our framework. We run the question "*Where was Michael Jack born?*" through the question encoder of DPR (Karpukhin et al., 2020), and project the question representation $\boldsymbol{e}_q$ to the vocabulary space using BERT's masked language modeling head (Devlin et al., 2019). The result is a distribution over the vocabulary, Q. We apply the same procedure for passages as well. These projections enable reasoning about and improving retrieval representations.

1 Introduction

Dense retrieval models based on neural text representations have proven very effective (Karpukhin et al., 2020; Qu et al., 2021; Ram et al., 2022; Izacard et al., 2022a,b), improving upon strong traditional sparse models like BM25 (Robertson and Zaragoza, 2009). However, when applied off-the-shelf (i.e., in *out-of-domain* settings) they often experience a severe drop in performance (Thakur et al., 2021; Sciavolino et al., 2021; Reddy et al., 2021). Moreover, the reasons for such failures are poorly understood, as the mechanism underlying dense representations remains under-investigated.

In this work, we present a new approach for interpreting and reasoning about dense retrievers, through distributions induced by their query[1] and passage representations when projected to the vocabulary space, namely distributions over their vocabulary space (Figure 1). Such distributions enable a better understanding of the representational nature of dense models and their failures, which paves the way to simple solutions that improve their performance.

We begin by showing that dense retrieval representations can be projected to the vocabulary space, by feeding them through the masked language modeling (MLM) head of the pretrained model they were initialized from *without any further training*.

[1]Throughout the paper, we use *query* and *question* interchangeably.

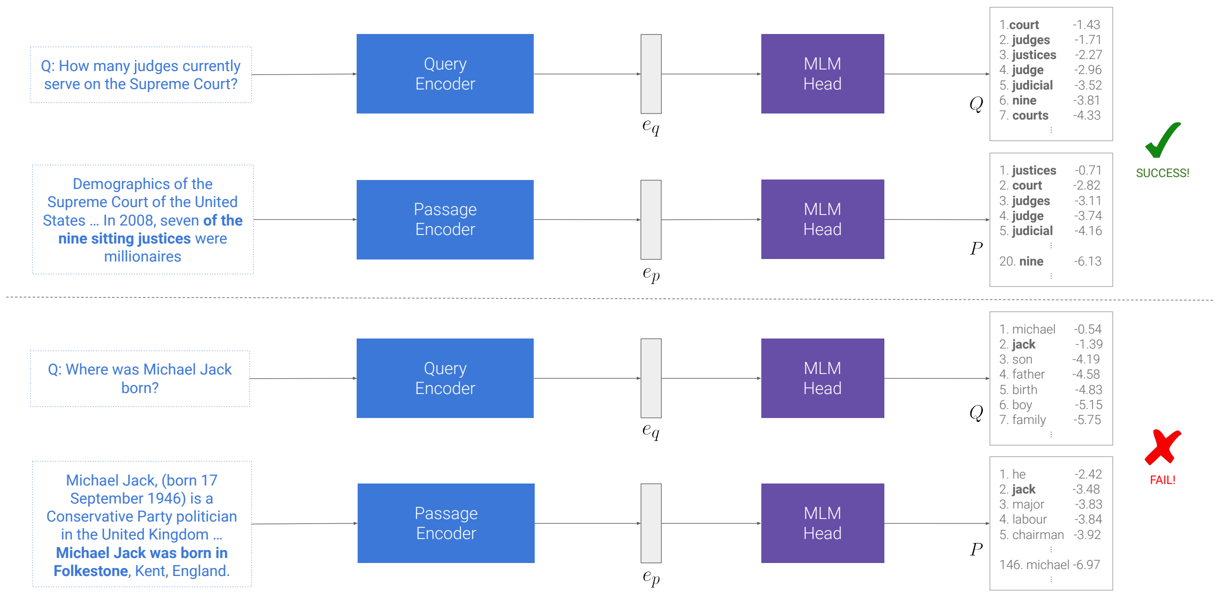


Figure 2: A success case from Natural Questions (top) and a failure case from EntityQuestions (bottom) of DPR (Karpukhin et al., 2020), explained via projecting question and (its relevant) passage representations to the vocabulary space. Tokens in the top-20 of both question and passage vocabulary projections are marked in bold.

This operation results in distributions over the vocabulary, which we refer to as *query vocabulary projections* and *passage vocabulary projections*.

Surprisingly, we find these projections to be highly interpretable to humans (Figure 2; Table 1). First, we analyze query projections and show that they contain both tokens that appear in the queries themselves, as well as additional tokens that are likely to appear in relevant passages. In other words, the model implicitly implements *query expansion* (Rocchio, 1971). For example, in Figure 2 the query is "*How many judges currently serve on the Supreme court?*", and the words in the query projection Q include "*justices*"—a synonym of "*judges*". An even more intriguing type of expansion is one where the added words actually contain information about the answer. For example, the word "*nine*" is also ranked high in Q even though it does not appear in the question itself, and nine is actually the correct answer.

We then continue to analyze passage projections, and show that they are likely to contain words that appear in queries about this passage. Thus, the passage projections can be viewed as anticipating the questions one would ask about the passage.

The above findings are especially surprising due to the fact that these retrieval models are fine-tuned in a contrastive fashion, and thus do not perform any prediction over the vocabulary or make any use of their language modeling head during fine-tuning. In addition, these representations are the result of running a deep transformer network that can implement highly complex functions. Nonetheless, model outputs remain "loyal" to the original lexical space learned during pretraining.

We further show that our approach is able to shed light on the reasons for which dense retrievers struggle with simple entity-centric questions (Sciavolino et al., 2021). Through the lens of vocabulary projections, we identify an interesting phenomenon: dense retrievers tend to "ignore" some of the tokens appearing in a given passage. This is reflected in the ranking assigned to such tokens in the passage projection. For example, the word "*michael*" in the bottom example of Figure 2 is ranked relatively low (even though it appears in the passage title), thereby hindering the model from retrieving this passage. We refer to this syndrome as *token amnesia*.

We leverage this insight and suggest a simple inference-time fix that enriches dense representations with lexical information, addressing token amnesia. We show that lexical enrichment significantly improves performance compared to vanilla models across multiple datasets. For example, we are able to boost the top-20 retrieval accuracy of DPR (Karpukhin et al., 2020) on the challenging EntityQuestions dataset (Sciavolino et al., 2021) by more than 15 points (from 49.7% to 65.4%).

Taken together, our analyses and results demonstrate the great potential of vocabulary projections as a framework for more principled research and development of dense retrieval models.

2 Background

In this work, we suggest a simple framework for interpreting dense retrieves, via projecting their representations to the vocabulary space. This is done using the (masked) language modeling head of their corresponding pretrained model. We now give the relevant background

2.1 Masked Language Modeling

Most language models based on encoder-only transformers (Vaswani et al., 2017) are pretrained using some variant of the masked language modeling (MLM) task (Devlin et al., 2019; Liu et al., 2019; Song et al., 2020), which involves masking some input tokens, and letting the model reconstruct them.

Specifically, for an input sequence $x_1, ..., x_n$, the transformer encoder is applied to output contextualized token representations $\boldsymbol{h}_1, ..., \boldsymbol{h}_n \in \mathbb{R}^d$. Then, to predict the missing tokens, an MLM head is applied to their contextualized representations. The MLM head is a function that takes a vector $\boldsymbol{h} \in \mathbb{R}^d$ as input and returns a distribution P over the model's vocabulary $\mathcal{V}$, defined as:

$$\text{MLM-Head}(\boldsymbol{h})[i] = \frac{\exp(\boldsymbol{v}_i^\top g(\boldsymbol{h}))}{\sum_{j\in\mathcal{V}} \exp(\boldsymbol{v}_j^\top g(\boldsymbol{h}))} \quad (1)$$

$g : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is a (potentially non-linear) function, e.g., a fully connected layer (followed by a LayerNorm for BERT (Devlin et al., 2019)), and $\boldsymbol{v}_i \in \mathbb{R}^d$ corresponds to the *static* embedding of the i-th item in the vocabulary.

2.2 Dense Retrieval

In dense retrieval, we are given a corpus of passages $\mathcal{C} = \{p_1, ..., p_m\}$ and a query q (e.g., a question or a fact to check), and we wish to compute query and passage representations ($\boldsymbol{e}_q$ and $\boldsymbol{e}_p$, respectively) such that similarity in this space implies high relevance of a passage to the query. Formally, let Enc_Q be a query encoder and Enc_P a passage encoder. These encoders are mappings from the input text to a vector in $\mathbb{R}^d$, and are obtained by fine-tuning a given LLM. Specifically, they return a pooled version of the LLM contextualized embeddings (e.g., the `[CLS]` embedding or mean pooling). We denote the embedding of the query and passage vectors as follows:

$$\begin{aligned} \boldsymbol{e}_q &= \text{Enc}_Q(q) \\ \boldsymbol{e}_p &= \text{Enc}_P(p) \end{aligned} \quad (2)$$

To fine-tune retrievers, a similarity measure $s(q, p)$ is defined (e.g., the dot-product between $\boldsymbol{e}_q$ and $\boldsymbol{e}_q$ or their cosine similarity) and the model is trained in a contrastive manner to maximize retriever accuracy (Lee et al., 2019; Karpukhin et al., 2020). Importantly, in this process, the MLM head function does not change at all.

3 Vocabulary Projections

We now describe our framework for projecting query and passage representations of dense retrievers to the vocabulary space. We utilize the MLM head, that maps from encoder output representations to distributions over the vocabulary (Eq. 1). Given a query q, we use the query encoder Enc_Q to obtain its representation $\boldsymbol{e}_q$ as in Eq. 2. Similarly, for a passage p we apply the passage encoder Enc_P to get $\boldsymbol{e}_p$. We then apply the MLM head as in Eq. (1) to obtain the vocabulary projection:

$$\begin{aligned} Q &= \text{MLM-Head}(\boldsymbol{e}_q) \\ P &= \text{MLM-Head}(\boldsymbol{e}_p) \end{aligned} \quad (3)$$

Note that it is not clear a-priori that Q and P will be meaningful in any way, as the encoder model has been changed since pretraining, while the MLM-head function remains fixed. Moreover, the MLM function has not been trained to decode "pooled" sequence-level representations (i.e., the results of CLS or mean pooling) during pretraining. Despite this intuition, we demonstrate here that P and Q are actually highly intuitive and can facilitate better understanding of dense retrievers.

4 Experiment Setup

To evaluate our framework and method quantitatively, we consider several dense retrieval models and datasets.

4.1 Models

We now list the retrievers used to demonstrate our framework and method. All dense models share the same architecture and size (i.e., that of BERT-base), and all were trained in a contrastive fashion with in-batch negatives—the prominent paradigm for training dense models (Lee et al., 2019; Karpukhin et al., 2020; Chang et al., 2020; Qu et al., 2021; Ram et al., 2022; Gao and Callan, 2022; Izacard et al., 2022a; Oguz et al., 2022; Ni et al., 2022; Chen et al., 2022).

Question	top-20 in Q	Passage	top-20 in P
where do the great lakes meet the ocean (A: the saint lawrence river)	lakes lake shore ocean confluence river water north canada meet east land rivers canoe sea border michigan connecting both shores	the great lakes , also called the laurent ##ian great lakes and the great lakes of north america , are a series of inter ##connected freshwater lakes located primarily in the upper mid - east region of north america , on the canada – united states border , which connect to the atlantic ocean through the saint lawrence river . they consist of lakes superior , michigan , huron ...	lakes lake the canada great freshwater water region ontario these central river rivers large basin core area erie all four
southern soul was considered the sound of what independent record label (A: motown)	southern music label soul motown blues nashville vinyl sound independent labels country records genre dixie record released gospel jazz south	soul music . the key sub ##gen ##res of soul include the detroit (motown) style , a rhythmic music influenced by gospel ; " deep soul " and " southern soul " , driving , energetic soul styles combining r & b with southern gospel music sound ; ... which came out of the rhythm and blues style ...	soul music jazz funk blues rock musical fusion genre black pure classical genres pop southern melody art like rich urban
who sings does he love me with re ##ba (A: linda davis)	duet song love music solo re he motown me his " pa album songs honey reprise bobby i peggy blues	" does he love you " is a song written by sandy knox and billy st ##rit ##ch , and recorded as a duet by american country music artists re ##ba mc ##ent ##ire and linda davis ...	he you him i it she his john we love paul who me does did yes why they how this

Table 1: Examples of questions and gold passages from the development set of Natural Questions, along with their 20 top-scored tokens in projections of DPR representations. Green tokens represent the lexical overlap signal (i.e., tokens that appear in both the question and the passage). Blue tokens represent query expansion (i.e., tokens that do not appear in the question but do appear in the passage.

DPR (Karpukhin et al., 2020) is a dense retriever that was trained on Natural Questions (Kwiatkowski et al., 2019). It was initialized from BERT-base (Devlin et al., 2019). Thus, we use the public pretrained MLM head of BERT-base to project DPR representations.

S-MPNet is a supervised model trained for Sentence Transformers (Reimers and Gurevych, 2019) using many available datasets for retrieval, sentence similarity, *inter alia*. It uses cosine similarity, rather than dot product, for relevance scores. It was initialized from MPNet-base (Song et al., 2020), and thus we use this model's MLM head.

Spider (Ram et al., 2022) is an unsupervised dense retriever trained using the *recurring span retrieval* pretraining task. It was also initialized from BERT-base, and we therefore use the same MLM head for projection as the one used for DPR.

BM25 (Robertson and Zaragoza, 2009) is a lexical model based on tf-idf. We use two variants of BM25: (1) vanilla BM25, and (2) BM25 over BERT/MPNet tokens (e.g., "*Reba*" → "*re ##ba*").[2]

4.2 Datasets

We follow prior work (Karpukhin et al., 2020; Ram et al., 2022) and consider six common open-domain question answering datasets for the evaluation of our framework: Natural Questions (NQ; Kwiatkowski et al. 2019), TriviaQA (Joshi et al., 2017), WebQuestions (WQ; Berant et al. 2013), CuratedTREC (TREC; Baudiš and Šedivý 2015), SQuAD (Rajpurkar et al., 2016) and EntityQuestions (EntityQs; Sciavolino et al. 2021).

4.3 Implementation Details

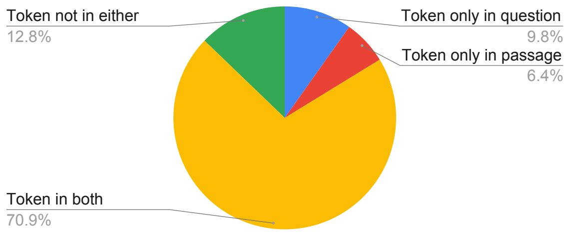
Our code is based on the official repository of DPR (Karpukhin et al., 2020), built on Hugging Face Transformers (Wolf et al., 2020).

We use the Wikipedia corpus standardized by Karpukhin et al. (2020), which uses the Wiki dump from Dec. 20, 2018. The corpus contains roughly 21 million passages of a hundred words each. For dense retrieval over this corpus, we apply exact search using FAISS (Johnson et al., 2021). For sparse retrieval we use Pyserini (Lin et al., 2021).

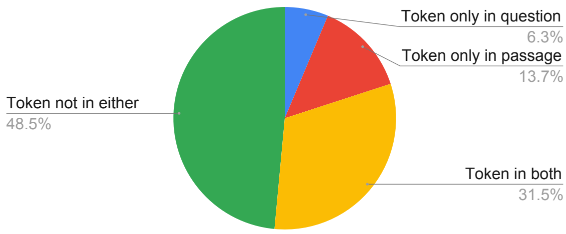
5 Vocabulary Projections: Analysis

In Section 3, we introduce a new framework for interpreting representations produced by dense retrievers. Next, we describe empirical findings that shed new light on what is encoded in these representations. Via vocabulary projections, we draw connections between dense retrievers and well-known concepts like *lexical overlap* and *lexical expansion*. We start by analyzing query representations (§5.1)

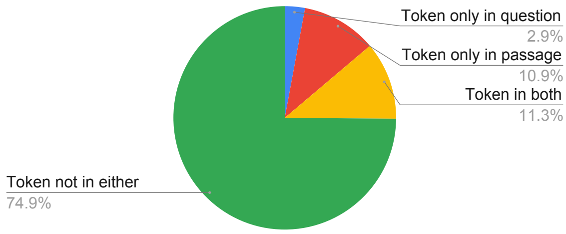
[2]BERT and MPNet use essentially the same vocabulary, up to special tokens.



(a) $k = 1$



(b) $k = 5$



(c) $k = 20$

Figure 3: An analysis of the top-k tokens in the vocabulary projection Q of DPR on questions in the development set of NQ. Specifically, we analyze whether these tokens are present in the question and/or in its gold positive passage. Similar analysis for P is given in Figure 6.

		Token-Level MRR in P		
		DPR	S-MPNet	Spider
Passage tokens	$\mathcal{T}_p$	0.03	0.03	0.02
Question tokens	$\mathcal{T}_q$	0.17	0.17	0.08
Only-in-q tokens	$\mathcal{T}_q \setminus \mathcal{T}_p$	0.02	0.01	0.01
Shared tokens	$\mathcal{T}_q \cap \mathcal{T}_p$	0.26	0.25	0.12

Table 2: An analysis of token-level mean reciprocal rank (MRR) in passage vocabulary projections P on the development set of NQ. For a pair of question q and its gold positive passage p, $\mathcal{T}_q$ and $\mathcal{T}_p$ are the corresponding sets of tokens, excluding stop words and punctuation marks. For each set $\mathcal{T}$, we report $\frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}} \frac{1}{\text{rank}_P(t)}$.

and then move to the analysis of passages (§5.2).

5.1 Projecting Query Representations

We begin our analysis by considering the top-scored items in vocabulary projections of queries. We ask: *are these words present in the question and/or its gold passage?* Answering this question may help draw conclusions on how a dense retriever builds its representations. For example, we wish to investigate to what extent are *lexical overlap* and *query expansion*, which are fundamental concepts in sparse retrieval, encoded in query representations. Consider the first example in Table 1. Two of the tokens shared by the question and passage ("lakes" and "ocean") are among the top-ranked in Q, therefore they are viewed as the *lexical overlap* signal. Also, six tokens from the passage *not present in the question* ("river", "north", "canada", "east", "border", "michigan") are also ranked high, thus we view them as *query expansion*. We now turn to quantify these phenomena.

To that end, Figure 3 analyzes the top-scored (i.e., top-k) tokens in Q, the vocabulary projections corresponding to the question. Specifically, for each question q and its corresponding gold passage p in the development set of NQ, we compute the question vocabulary projection Q, and report the overlap between top-k tokens in Q, and the tokens in q and p. Overall, it is evident that the tokens of q and p constitute a major part of the top-scored tokens in Q (the union of the blue, red and yellow slices), and the top-scored token (Figure 3a) is present in at least one of them in 87.2% of the questions. This is evidence that these tokens are meaningful with respect to both q and p.

Lexical Overlap Tokens shared by q and p constitute the *lexical overlap* signal in retrieval, used by sparse methods like BM25. *How prominent are they in question projections?* Figure 3 illustrates that for more than 70% of the questions, the first token in Q *appears in both* the question q and its gold passage p (yellow slice). When considering the top-5 tokens in Q, more than 30% of the tokens are shared tokens. These findings emphasize that even for dense retrievers—which do not operate at the lexical level—lexical overlap remains a highly dominant signal.

Query Expansion To overcome the "vocabulary mismatch" problem (i.e., when question-document pairs are semantically relevant, but lack significant lexical overlap), *query expansion* methods have been studied extensively (Rocchio, 1971; Voorhees, 1994; Zhao and Callan, 2012; Mao et al., 2021). The main idea is to expand the query with additional terms that will better guide the retrieval process. Figure 3 suggests that query expansion is also dominant in dense retrieval. We define a token as a query expansion if it does not appear in the query itself but does appear in the query projection Q, and also in the gold passage of that query p. Surprisingly, almost 14% of the tokens in the top-5 of Q are query expansion tokens (red slices). We note that there are two interesting classes of query expansion tokens: (1) synonyms of question tokens, as well as tokens that share similar semantics with tokens in q (e.g., "michigan" in the first example of Table 1). (2) "answer tokens" which contain the answer to the query (e.g., "motown" in the second example of Table 1). The presence of such tokens may suggest the model already "knows" the answer to the given question, either from pretraining or from similar questions seen during training.

5.2 Projecting Passage Representations

We continue our projection-based analysis of dense retrievers by asking, *what is encoded in passage representations?* We show that vocabulary projections can provide intuitive explanations and discover interesting phenomena concerning this question as well.

Encoding passages is a fundamentally different task than encoding questions, as the former contain significantly more information than the latter.[3] Intuitively, a passage representation should focus on information that is likely to be asked about. We now investigate whether this intuition is manifested in passage vocabulary projections.

Query Prediction We start our passage-side analysis by hypothesizing that out of passage tokens, *those that are likely to appear in relevant questions receive higher scores than others*. To test our hypothesis, we analyze the ranks of question and passage tokens in passage vocabulary projections, P. Formally, let $\mathcal{T}_q$ and $\mathcal{T}_p$ be the *sets* of tokens in a question q and its gold passage p, respectively. Table 2 exhibits the token-level mean reciprocal rank (MRR) of these sets in P. We observe that tokens shared by q and p (i.e., $\mathcal{T}_q \cap \mathcal{T}_p$) are ranked significantly higher than other passage tokens (i.e., $\mathcal{T}_p$). For example, in DPR the MRR of shared tokens is 0.26, while that of other passage tokens is only 0.03. This trend holds for other models as well. This finding thus supports our claim and quantifies that tokens that appear in relevant questions are ranked higher than others. This behavior suggests that the model *predicts* which of the input passage tokens will appear in likely questions.

Additionally, the results in Table 2 suggest an intuitive explanation to why unsupervised dense retrievers like Spider still lag behind supervised ones as DPR and S-MPNet (especially in top-k accuracy for lower values of k, cf. Spider results in Table 4): unsupervised models are dramatically less effective in query prediction (e.g., the MRR of shared tokens in Spider is 0.12, compared to 0.26 and 0.25 in DPR and S-MPNet). This might be due to the fact that unsupervised models do not observe real questions in their training (e.g., Spider synthesizes pseudo-questions on real passages), and are thus less effective in query prediction.

(Lack of) Document Expansion In §5.1, we argue that dense models implicitly apply query expansion to drive their retrieval process. However, expanding documents (or passages) is also a popular approach for improving retrieval (Tao et al., 2006; Efron et al., 2012; Nogueira et al., 2019; MacAvaney et al., 2020). *Is document expansion used by dense retrievers?* We find signals suggesting this is *not* the case. Table 2 indicates that tokens from $\mathcal{T}_q \setminus \mathcal{T}_p$ (i.e., tokens that appear in q but not in p) are not frequent among the top-scored tokens in P, as their MRR values is particularly low, 0.01–0.02. Also, Figure 6 demonstrates that these tokens constitute only 1% of the top-k tokens in P (blue slices). We hypothesize that since there are many questions that can be asked for a single passage, anticipating the relevant lexical expansion is challenging.

6 Token Amnesia

Dense retrievers have shown difficulties in *out-of-domain* settings (Sciavolino et al., 2021; Thakur et al., 2021), where even sparse models like BM25 significantly outperform them. These findings point at a shortcoming of dense retrievers: They are less effective in lexical representation of questions and passages than bag-of-words models. For example, in Figure 2 the token "*michael*" is missing from the top-k of the passage projection P.

Next, we delve deeper into this phenomenon through the framework of vocabulary projections. We demonstrate that failures of dense retrievers highly correlate with poor representation of input tokens in the model's vocabulary projection (§6.1). To overcome this issue, we suggest a lexical enrichment procedure for dense representations (§6.2)

[3] The average question length in Natural Questions (using BERT's vocabulary) is 9.8 tokens, while the average passage length in our corpus is 135 tokens.

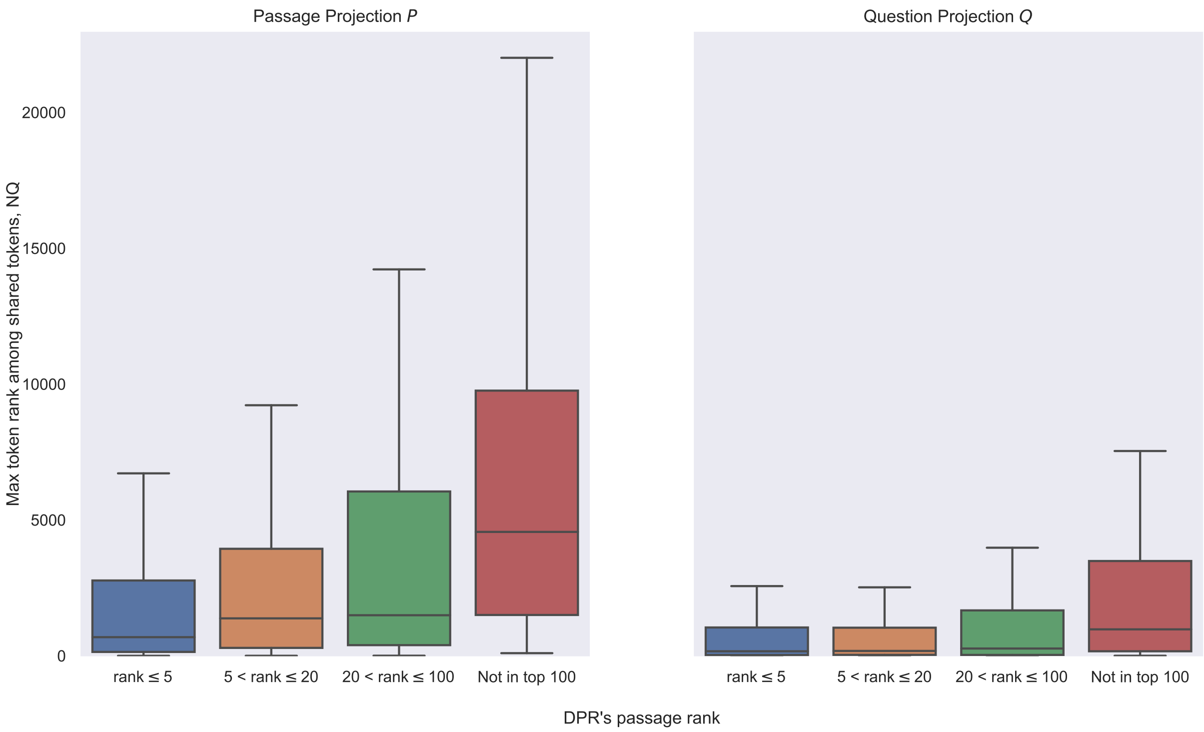


Figure 4: An analysis of *token amnesia*. We consider questions for which BM25 retrieves a correct passage (i.e., a passage that contains the answer) in its top-5, and analyze what ranks were assigned to tokens *shared* by the question and the passage in the passage vocabulary projection P (left) and question vocabulary projection Q (right). We plot the maximal token rank as a function of the rank assigned to the correct passage by DPR.

and demonstrate its effectiveness on downstream retrieval performance (§6.3). We conclude with an ablation study(§6.4).

6.1 Analysis

We now offer an intuitive explanation to these failures using our *vocabulary projections* framework. We focus on setups where BM25 outperforms dense models and ask: *why do dense retrievers fail to model lexical overlap signals?* To answer this question, we consider subsets of NQ and EntityQs where BM25 succeeds retrieves a correct passage in its top-5 results. We focus on these subsets as they contain significant lexical overlap between questions and passages by definition.

Let q be a question and p the positive passage retrieved by BM25 for q, and Q and P be their corresponding vocabulary projections for some dense retriever. Also, let $\mathcal{T} \subseteq \mathcal{V}$ be the set of tokens that appear in both q and p (excluding stop words). Figure 4 shows the maximum (i.e., lowest) rank of tokens from $\mathcal{T}$ in the distributions P (left) and Q (right) as a function of whether DPR is able to retrieve this passage (i.e., the rank of p in the retrieval results of DPR). Indeed, the median max-rank over questions for which DPR succeeds to fetch p in its top-5 results (blue box) is 686 (in other words, for more than half of these questions, all shared tokens are ranked in the top-1000 in P). However, when DPR fails (red box), we observe much higher ranks for the shared tokens (median is 4561, upper quartile is 9760). As expected (due to the fact that questions contain less tokens than passages), the ranks of shared tokens in question projections Q are much lower. However, the trend is present in the question side as well. Additional figures (for EntityQs; as well as median ranks instead of max ranks) are given in App. B.

These findings suggest that dense retrievers tend to "ignore" tokens that appear in their inputs, a behavior which we refer to as *token amnesia*. This phenomenon can also be connected to query prediction (§5.2): dense retrievers fail to predict query tokens when query distribution changes. We argue that this behavior is the reason for their relatively-poor performance in out-of-domain scenarios.

Next, we introduce a method to address token amnesia in dense retrievers, via lexical enrichment of dense representations.

Model	EntityQs	TriviaQA	WQ	TREC	SQuAD
BM25	71.4	76.4	62.4	81.1	71.2
BM25 (BERT/MPNet Vocabulary)	66.2	75.8	62.1	79.3	70.0
DPR	49.7	69.0	68.8	85.9	48.9
DPR + LE	**65.4**	**75.3**	**73.2**	**87.9**	**59.7**
S-MPNet	57.6	77.6	73.9	90.2	65.5
S-MPNet + LE	**68.5**	**78.9**	**74.5**	**90.4**	**69.0**
Spider	66.3	75.8	65.9	82.6	61.0
Spider + LE	**68.9**	**76.3**	**70.2**	**83.4**	**62.8**

Table 3: Top-20 retrieval accuracy in a "zero-shot" setting (i.e., datasets were not used for model training). LE stands for *lexical enrichment* (our method; §6.2), that enriches query and passage representation with lexical information. BM25 (BERT Vocabulary) refers to a model that operates over tokens from BERT's vocabulary, rather than words. For each model and dataset, we compare the enriched (LE) model with the original, and mark in bold the better one from the two. We underline the best overall model for each dataset. Top-$\{1, 5, 100\}$ accuracy is given in Tables 6, 7, 8.

Model	NQ			
	Top-1	Top-5	Top-20	Top-100
DPR	**46.3**	**68.3**	80.1	86.1
DPR + LE	45.8	67.8	**80.8**	**87.2**
S-MPNet	37.7	**66.4**	**80.9**	87.8
S-MPNet + LE	**39.3**	66.0	80.2	**88.0**
Spider	24.8	49.6	**68.3**	81.2
Spider + LE	**25.9**	**50.4**	67.8	**81.4**

Table 4: Top-k retrieval accuracy of different models on the test set of NQ, with and without lexical enrichment (LE). NQ was used for training DPR and S-MPNet.

6.2 Method: Lexical Enrichment

As suggested by the analysis in §6.1, dense retrievers have the tendency to ignore some of their input tokens. We now leverage this insight to improve these models. We refer to our method as *lexical enrichment* (LE) because it enriches text encodings with specific lexical items.

Intuitively, a natural remedy to the "token amnesia" problem is to change the retriever encoding such that *it does* include these tokens. For example, assume the query q is "*Where was Michael Jack born?*" and the corresponding passage p contains the text "*Michael Jack was born in Folkestone, England*". According to Figure 2, the token "*michael*" is ranked relatively low in P, and DPR fails to retrieve the correct passage p. We would like to modify the passage representation $\boldsymbol{e}_p$ and get an *enriched* version $\boldsymbol{e}'_p$ that does have this token in its top-k projected tokens, while keeping most of the other projected tokens intact. This is our goal in LE, and we next describe the approach. We focus on enrichment of passage representations, as query enrichment works similarly. We first explain how to enrich representations with a single token, and then extend the process to multiple tokens.

Single-Token Enrichment Assume we want to enrich a passage representation $\boldsymbol{e}_p$ with a token t (e.g., $t =$ "*michael*" in the above example). If there were no other words in the passage, we'd simply want to find an embedding such that feeding it into the MLM would produce t as the top token.[4] We refer to this embedding as the *single-token enrichment* of t, denote it by $\boldsymbol{s}_t$ and define it as:[5]

$$\boldsymbol{s}_t = \arg\max_{\hat{\boldsymbol{s}}} \log \mathrm{MLM}(\hat{\boldsymbol{s}})[t] \quad (4)$$

In order to solve the optimization problem in Eq. 4 for each t in the vocabulary, we use Adam with a learning rate of 0.01.[6] We stop when a (cross-entropy) loss threshold of 0.1 is reached for all tokens. We then apply whitening (Jung et al., 2022), which has proven effective for dense retrieval.

Multi-Token Enrichment Now suppose we have an input x (either a question or a passage) and we'd like to enrich its representation with its tokens $x = [x_1, .., x_n]$, such that rare tokens are given higher weights than frequent ones (as in BM25). Then, we simply take its original representation $\boldsymbol{e}_x$ and add to it a weighted sum of the single-token

[4] Note that feeding the token input embedding $\boldsymbol{v}_t$ does not necessarily produce t as the top token, as the MLM head applies a non-linear function g (Eq. 1).

[5] This is equivalent to the cross-entropy loss between a one-hot vector on t and the output distribution $MLM(\hat{\boldsymbol{s}})$.

[6] For S-MPNet, we used a learning rate of 10^{-3}.

Method	NQ (Dev Set)				EntityQs (Dev Set)			
	Top-1	**Top-5**	**Top-20**	**Top-100**	**Top-1**	**Top-5**	**Top-20**	**Top-100**
DPR	44.9	66.8	78.1	85.0	24.0	38.4	50.4	63.5
DPR + LE	44.4	67.5	**79.4**	**86.0**	**38.3**	**54.0**	**65.2**	**76.1**
No IDF	**45.1**	67.3	78.5	85.4	32.0	46.4	57.7	69.6
BERT embedding matrix	44.8	**67.6**	79.1	85.6	34.6	50.3	61.8	72.8
No whitening	44.1	66.3	78.7	85.2	34.6	49.7	61.4	72.9
No ℓ_2 normalization	43.9	66.8	79.2	**86.0**	35.5	51.3	63.0	74.6

Table 5: Ablation study on the development set of Natrual Questions and Entity Questions. DPR + LE is our lexical enrichment method applied on DPR. *No IDF* removes the IDF weights in Eq. 5 (i.e., mean pooling). *BERT embedding matrix* replaces sinle-token enrichment $\boldsymbol{s}_t$ as defined in Eq. 4 with the static token embeddings of BERT, $\boldsymbol{v}_t$ (Eq. 1). *No whitening* removes whitening transformation. *No ℓ_2 normalization* removes the normalization of $\boldsymbol{e}_x^{\text{lex}}$.

representations (Eq.4). Namely, we define:

$$\begin{aligned} \boldsymbol{e}_x^{\text{lex}} &= \frac{1}{n}\sum_{i=1}^{n} w_{x_i}\boldsymbol{s}_{x_i} \\ \boldsymbol{e}'_x &= \boldsymbol{e}_x + \lambda \cdot \frac{\boldsymbol{e}_x^{\text{lex}}}{||\boldsymbol{e}_x^{\text{lex}}||} \end{aligned} \tag{5}$$

Here λ is a hyper-parameter chosen via cross validation. We use the inverse document frequency (Sparck Jones, 1972) of tokens as their weights: $w_{x_i} = \text{IDF}(x_i)$. The relevance score is then defined on the enriched representations: $s(q,p) = \boldsymbol{e}'^{\top}_q \boldsymbol{e}'_p$.

6.3 Results

Our experiments demonstrate the effectiveness of our method for multiple models, especially in zero-shot settings. Table 3 shows the top-20 accuracy of several models with and without our enrichment method, LE. Results for top-1, top-5 and top-100 accuracy are given in App. C. The results demonstrate the effectiveness of LE when added to all baseline models. For example, our refinement procedure is able to boost DPR performance on EntityQuestions by more than 15 points. Significant improvements are observed on EntityQs using LE for S-MPNet and Spider as well (10.9% and 2.6% absolute improvements, respectively). In addition, the two latter models obtain higher accuracy than BM25 that operates on the same units (i.e., BM25 with BERT vocabulary). This finding indicates that they are able to integrate semantic information (from the original representation) with lexical signals. Yet, vanilla BM25 is still better than LE models on EntityQs and SQuAD, which prompts further work on how to incorporate lexical signal in dense retrieval. Overall, it is evident that LE improves retrieval accuracy compared with baseline models for all models and unseen datasets (i.e., zero-shot setting), while keeping roughly the same accuracy on NQ (Table 4).

6.4 Ablation Study

We carry an ablation study, to test our design choices from §6.2. We evaluate four elements of our method: (1) The use of IDF to highlight rare tokens, (2) Our approach for deriving single-token representations, (3) The use of whitening, and (4) The use of unit normalization.

IDF In our method, we create lexical representations of questions and passages, $\boldsymbol{e}_x^{\text{lex}}$. These lexical representations are the average of token embeddings, each multiplied by its token's IDF. We validate that IDF is indeed necessary – Table 5 demonstrates that setting $w_{x_i} = 1$ in Eq. 5 (i.e., mean pooling over single-token representations) leads to a significant degradation in performance on EntityQs. For example, top-20 retrieval accuracy drops from 65.2% to 57.7%.

Single-Token Enrichment Eq. 4 defines how we create single-token representations for our vocabulary items: for each item in the vocabulary $v \in \mathcal{V}$, we find an embedding which gives a one-hot vector peaked at v when fed to the MLM head. The main intuition here is that we want these embeddings to be aligned with the retrieval representation space. We confirm this intuition by replacing Eq. 4 with the static embeddings of the pretrained model (e.g., BERT in the case of DPR). We find that our approach significantly improves over BERT's embeddings on EntityQs (e.g., the margin in top-20 accuracy is 3.4%).

Whitening & Normalization Last, we experiment with removing the whitening and ℓ_2 normalization. It is evident that they are both necessary, as removing either of them causes a dramatic drop in performance (3.8% and 2.2% in top-20 accuracy on EntityQs, respectively).

7 Related Work

Projecting representations and model parameters to the vocabulary space has been studied previously mainly in the context of language models. Geva et al. (2021) showed that feed-forward layers in transformers can be regarded as key-value memories, where the value vectors induce distributions over the vocabulary. Geva et al. (2022) view the token representations themselves as inducing such distributions, with feed-forward layers "updating" them. Dar et al. (2022) suggest to project all transformer parameters to the vocabulary space. Dense retrieval models, however, do not have any language modeling objective during fine-tuning, yet we show that their representations can still be projected to the vocabulary.

Despite the wide success of dense retrievers recently, interpreting their representations remains under-explored. MacAvaney et al. (2022) analyze neural retrieval models (not only dense retrievers) via diagnostic probes, testing characteristics like sensitivity to paraphrases, styles and factuality. Adolphs et al. (2022) decode the query representations of neural retrievers using a T5 decoder, and show how to "move" in representation space to decode better queries for retrieval.

Language models (and specifically MLMs) have been used for *sparse retrieval* in the context of term-weighting and lexical expansion. For example, Bai et al. (2020) and Formal et al. (2021) learn such functions over BERT's vocabulary space. We differ by showing that *dense retrievers* implicitly operate in that space as well. Thus, these approaches may prove effective for dense models as well. While we focus in this work on dense retrievers based on encoder-only models, our framework is easily extendable for retrievers based on autoregressive decoder-only (i.e., left-to-right) models like GPT (Radford et al., 2019; Brown et al., 2020), e.g., Neelakantan et al. (2022) and Muennighoff (2022).

The dual encoder architecture is used for other tasks other than retrieval. For example, it is highly popular for semantic similarity (Reimers and Gurevych, 2019; Gao et al., 2021), zero-shot entity linking (Logeswaran et al., 2019; Wu et al., 2020), *inter alia*. While we focus on interpreting dense retrievers, our framework can be easily deployed for such models as well. In addition, we envision applications of vocabulary projections for models that build explicit question and phrase representations for question answering and phrase retrieval (Ram et al., 2021; Lee et al., 2021).

8 Conclusion

In this work, we explore projecting query and passage representations obtained by dense retrieval to the vocabulary space. We show that these projections help facilitate a better understanding of the mechanisms underlying dense retrieval, as well as their failures. We also demonstrate how projections can help improve these models. This understanding is likely to help in improving retrievers, as our lexical enrichment approach demonstrates.

Acknowledgements

We thank Ori Yoran, Yoav Levine, Yuval Kirstain and Mor Geva for valuable feedback and discussions. This project was funded by the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant ERC HOLI 819080), the Blavatnik Fund, the Alon Scholarship, the Yandex Initiative for Machine Learning, Intel Corporation, ISRAEL SCIENCE FOUNDATION (grant No. 448/20), Open Philanthropy, and an Azrieli Foundation Early Career Faculty Fellowship.

References

Leonard Adolphs, Michelle Chen Huebscher, Christian Buck, Sertan Girgin, Olivier Bachem, Massimiliano Ciaramita, and Thomas Hofmann. 2022. Decoding a neural retriever's latent space for query suggestion.

Yang Bai, Xiaoguang Li, Gang Wang, Chaoliang Zhang, Lifeng Shang, Jun Xu, Zhaowei Wang, Fangshan Wang, and Qun Liu. 2020. SparTerm: Learning term-based sparse representation for fast text retrieval.

Petr Baudiš and Jan Šedivý. 2015. Modeling of the question answering task in the YodaQA system. In *Proceedings of the 6th International Conference on Experimental IR Meets Multilinguality, Multimodality, and Interaction - Volume 9283*, CLEF'15, page 222–228, Berlin, Heidelberg. Springer-Verlag.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*.

Wei-Cheng Chang, Felix X. Yu, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. 2020. Pre-training tasks for embedding-based large-scale retrieval. In *8th International Conference on Learning Representations, ICLR 2020*.

Xilun Chen, Kushal Lakhotia, Barlas Oğuz, Anchit Gupta, Patrick Lewis, Stan Peshterliev, Yashar Mehdad, Sonal Gupta, and Wen-tau Yih. 2022. Salient phrase aware dense retrieval: Can a dense retriever imitate a sparse one?

Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. 2022. Analyzing transformers in embedding space.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Miles Efron, Peter Organisciak, and Katrina Fenlon. 2012. Improving retrieval of short texts through document expansion. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, page 911–920, New York, NY, USA. Association for Computing Machinery.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. SPLADE: Sparse lexical and expansion model for first stage ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2288–2292, New York, NY, USA. Association for Computing Machinery.

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. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022a. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022b. Atlas: Few-shot learning with retrieval augmented language models.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Euna Jung, Jungwon Park, Jaekeol Choi, Sungyoon Kim, and Wonjong Rhee. 2022. Isotropic representation can improve dense retrieval.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. 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.

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. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Jinhyuk Lee, Mujeen Sung, Jaewoo Kang, and Danqi Chen. 2021. Learning dense representations of phrases at scale. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6634–6647, Online. Association for Computational Linguistics.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

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*, SIGIR '21, page 2356–2362, New York, NY, USA. Association for Computing Machinery.

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.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy. Association for Computational Linguistics.

Sean MacAvaney, Sergey Feldman, Nazli Goharian, Doug Downey, and Arman Cohan. 2022. ABNIRML: Analyzing the behavior of neural IR models. *Transactions of the Association for Computational Linguistics*, 10:224–239.

Sean MacAvaney, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, Nazli Goharian, and Ophir Frieder. 2020. Expansion via prediction of importance with contextualization. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1573–1576, New York, NY, USA. Association for Computing Machinery.

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Generation-augmented retrieval for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4089–4100, Online. Association for Computational Linguistics.

Niklas Muennighoff. 2022. Sgpt: Gpt sentence embeddings for semantic search.

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. 2022. Text and code embeddings by contrastive pre-training.

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. 2022. Large dual encoders are generalizable retrievers.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*.

Barlas Oguz, Kushal Lakhotia, Anchit Gupta, Patrick Lewis, Vladimir Karpukhin, Aleksandra Piktus, Xilun Chen, Sebastian Riedel, Scott Yih, Sonal Gupta, and Yashar Mehdad. 2022. Domain-matched pre-training tasks for dense retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1524–1534, Seattle, United States. Association for Computational Linguistics.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Ori Ram, Yuval Kirstain, Jonathan Berant, Amir Globerson, and Omer Levy. 2021. Few-shot question answering by pretraining span selection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3066–3079, Online. Association for Computational Linguistics.

Ori Ram, Gal Shachaf, Omer Levy, Jonathan Berant, and Amir Globerson. 2022. Learning to retrieve passages without supervision. In *Proceedings of the*

2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2687–2700, Seattle, United States. Association for Computational Linguistics.

Revanth Gangi Reddy, Vikas Yadav, Md Arafat Sultan, Martin Franz, Vittorio Castelli, Heng Ji, and Avirup Sil. 2021. Towards robust neural retrieval models with synthetic pre-training.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Joseph Rocchio. 1971. Relevance feedback in information retrieval. *The SMART retrieval system: experiments in automatic document processing*, pages 313–323.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6138–6148, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*, volume 33, pages 16857–16867. Curran Associates, Inc.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. In *Journal of Documentation*, volume 28 no. 1, pages 11–21.

Tao Tao, Xuanhui Wang, Qiaozhu Mei, and ChengXiang Zhai. 2006. Language model information retrieval with document expansion. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 407–414, New York City, USA. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008.

Ellen M. Voorhees. 1994. Query expansion using lexical-semantic relations. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, page 61–69, Berlin, Heidelberg. Springer-Verlag.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.

Le Zhao and Jamie Callan. 2012. Automatic term mismatch diagnosis for selective query expansion. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, page 515–524, New York, NY, USA. Association for Computing Machinery.

A Analysis: Further Results

Figure 6 gives an analysis of the top-k tokens in passage projections P (complements Figure 3).

B Token Amnesia: Further results

Figure 5 gives further analyses of token amnesia: It contains the results for EntityQuestions, as well as analysis of median ranks in addition to max ranks (complements Figure 4).

C Lexical Enrichment: Further Results

Table 6, Table 7 and Table 8 give the zero-shot results of adding lexical enrichment to dense models for $k \in \{1, 5, 100\}$, respectively (complement Table 3).

Model	EntityQs	TriviaQA	WQ	TREC	SQuAD
BM25	43.5	46.3	18.9	34.6	36.7
BM25 (BERT/MPNet Vocabulary)	37.6	45.4	19.2	33.0	35.6
DPR	24.3	37.3	30.5	51.3	16.0
DPR + LE	**38.3**	**45.8**	**35.0**	**54.6**	**22.8**
S-MPNet	22.7	42.9	30.9	51.0	25.8
S-MPNet + LE	**37.3**	**47.3**	**37.1**	**54.0**	**30.0**
Spider	35.0	41.7	22.3	38.2	22.2
Spider + LE	**40.7**	**43.7**	**27.8**	**43.2**	**23.5**

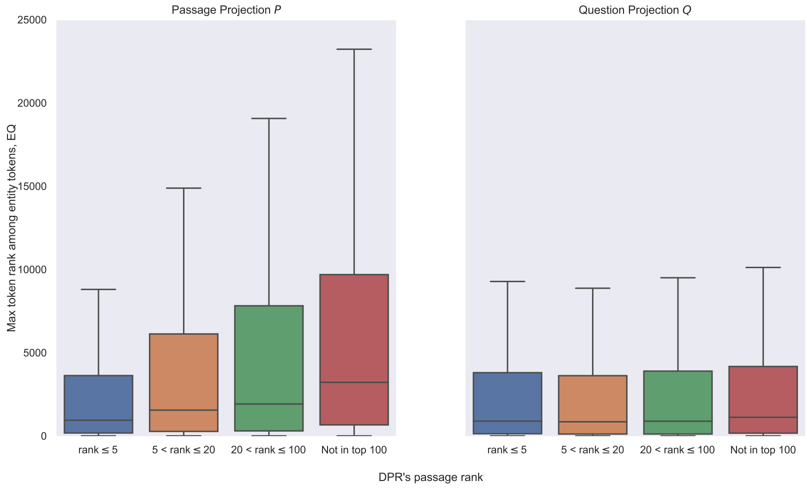
Table 6: Top-1 retrieval accuracy in a "zero-shot" setting (i.e., datasets were not used for model training), complementary to Table 3. LE stands for *lexical enrichment* (our method; §6.2), that enriches query and passage representation with lexical information. BM25 (BERT Vocabulary) refers to a model that operates over tokens from BERT's vocabulary, rather than words. For each model and dataset, we compare the enriched (LE) model with the original, and mark in bold the better one from the two. We underline the best overall model for each dataset.

Model	EntityQs	TriviaQA	WQ	TREC	SQuAD
BM25	61.0	66.3	41.8	64.6	57.5
BM25 (BERT/MPNet Vocabulary)	55.1	65.6	42.3	62.5	56.1
DPR	38.1	57.0	52.7	74.1	33.4
DPR + LE	**53.8**	**64.8**	**57.7**	**79.5**	**42.3**
S-MPNet	42.7	66.1	58.8	79.7	49.5
S-MPNet + LE	**56.8**	**68.5**	**61.6**	**81.4**	**53.2**
Spider	54.5	63.6	46.8	65.9	43.6
Spider + LE	**58.0**	**64.4**	**52.2**	**70.0**	**44.9**

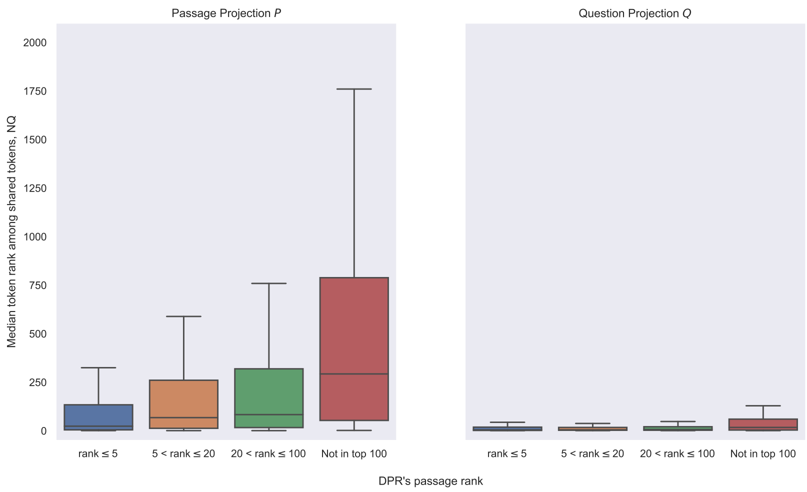
Table 7: Top-5 retrieval accuracy in a "zero-shot" setting (i.e., datasets were not used for model training), complementary to Table 3. LE stands for *lexical enrichment* (our method; §6.2), that enriches query and passage representation with lexical information. BM25 (BERT Vocabulary) refers to a model that operates over tokens from BERT's vocabulary, rather than words. For each model and dataset, we compare the enriched (LE) model with the original, and mark in bold the better one from the two. We underline the best overall model for each dataset.

Model	EntityQs	TriviaQA	WQ	TREC	SQuAD
BM25	80.0	83.2	75.5	90.3	82.0
BM25 (BERT/MPNet Vocabulary)	76.6	83.0	76.0	90.5	81.1
DPR	63.2	78.7	78.3	92.1	65.1
DPR + LE	**76.1**	**82.9**	**82.1**	**93.5**	**74.0**
S-MPNet	71.7	84.8	83.0	**95.1**	78.4
S-MPNet + LE	**78.6**	**85.1**	**83.8**	95.0	**80.7**
Spider	77.4	83.5	79.7	**92.8**	76.0
Spider + LE	**78.9**	**83.8**	**81.5**	92.2	**77.8**

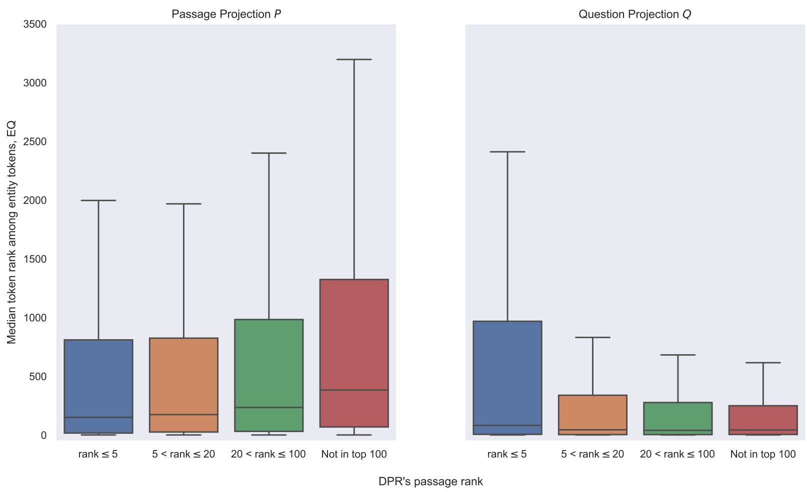
Table 8: Top-100 retrieval accuracy in a "zero-shot" setting (i.e., datasets were not used for model training), complementary to Table 3. LE stands for *lexical enrichment* (our method; §6.2), that enriches query and passage representation with lexical information. BM25 (BERT Vocabulary) refers to a model that operates over tokens from BERT's vocabulary, rather than words. For each model and dataset, we compare the enriched (LE) model with the original, and mark in bold the better one from the two. We underline the best overall model for each dataset.



(a) Max rank among shared tokens, EntityQuestions

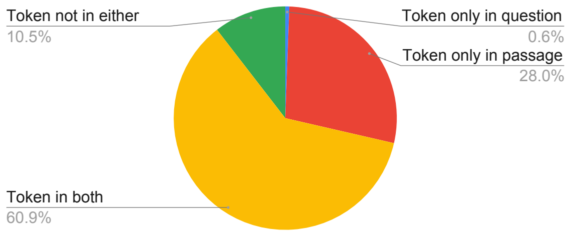


(b) Median rank among shared tokens, Natural Questions

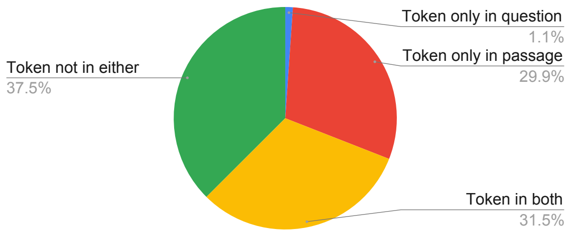


(c) Median rank among shared tokens, EntityQuestions

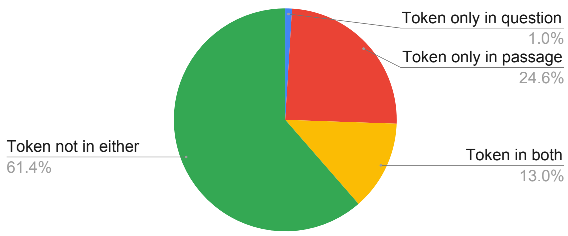
Figure 5: Further analysis of *token amnesia* (complementary to Figure 4). We consider questions for which BM25 retrieves a correct passage (i.e., a passage that contains the answer) in its top-5, and analyze what ranks were assigned to tokens *shared* by the question and the passage in the passage vocabulary projection P (left) and question vocabulary projection Q (right). We plot the max and median token rank as a function of the rank assigned to the correct passage by DPR, for Natural Questions (NQ) and EntityQuestions (EQ).



(a) $k = 1$



(b) $k = 5$



(c) $k = 20$

Figure 6: An analysis of the top-k tokens in the vocabulary projection P of DPR on gold passages of questions in the development set of NQ. Specifically, we analyze whether these tokens are present in the gold passage and/or in its corresponding question. Similar plots for Q are given in Figure 3.