Sparse, Dense, and Attentional Representations for Text Retrieval

Yi Luan*, **Jacob Eisenstein***, **Kristina Toutanova***, **Michael Collins**

Google Research
{luanyi, jeisenstein, kristout, mjcollins}@google.com

Abstract

Dual encoders perform retrieval by encoding documents and queries into dense low-dimensional vectors, scoring each document by its inner product with the query. We investigate the capacity of this architecture relative to sparse bag-of-words models and attentional neural networks. Using both theoretical and empirical analysis, we establish connections between the encoding dimension, the margin between gold and lower-ranked documents, and the document length, suggesting limitations in the capacity of fixed-length encodings to support precise retrieval of long documents. Building on these insights, we propose a simple neural model that combines the efficiency of dual encoders with some of the expressiveness of more costly attentional architectures, and explore sparse-dense hybrids to capitalize on the precision of sparse retrieval. These models outperform strong alternatives in large-scale retrieval.

1 Introduction

Retrieving relevant documents is a core task for language technology, and is a component of applications such as information extraction and question answering (e.g., Narasimhan et al., 2016; Kwok et al., 2001; Voorhees, 2001). While classical information retrieval has focused on heuristic weights for sparse bag-of-words representations (Spärck Jones, 1972), more recent work has adopted a two-stage retrieval and ranking pipeline, where a large number of documents are retrieved using sparse high dimensional query/document representations, and are further reranked with learned neural models (Mitra and Craswell, 2018). This two-stage approach has achieved state-of-the-art results on IR benchmarks (Nogueira and Cho, 2019; Yang et al., 2019; Nogueira et al., 2019a), especially since sizable annotated data has become available for training deep neural models (Dietz et al., 2018; Craswell et al., 2020). However, this pipeline suffers from a strict upper bound imposed by any recall errors in the first-stage retrieval model: For example, the recall@1000 for BM25 reported by Yan et al. (2020) is 69.4.

A promising alternative is to perform first-stage retrieval using learned dense low-dimensional encodings of documents and queries (Huang et al., 2013; Reimers and Gurevych, 2019; Gillick et al., 2019; Karpukhin et al., 2020). The dual encoder model scores each document by the inner product between its encoding and that of the query. Unlike full attentional architectures, which require extensive computation on each candidate document, the dual encoder can be easily applied to very large document collections thanks to efficient algorithms for inner product search; unlike untrained sparse retrieval models, it can exploit machine learning to generalize across related terms.

To assess the relevance of a document to an information-seeking query, models must *both* (i) check for precise term overlap (for example, presence of key entities in the query) and (*ii*) compute semantic similarity generalizing across related concepts. Sparse retrieval models excel at the first sub-problem, while learned dual encoders can be better at the second. Recent history in NLP might suggest that learned dense representations should always outperform sparse features overall, but this is not necessarily true: as shown in Figure 1, the BM25 model (Robertson et al., 2009) can outperform a dual encoder based on BERT, particularly on longer documents and on a task that requires precise detection of word overlap.[1] This raises questions about the limitations of dual

*Equal contribution.

[1]See §4 for experimental details.

Transactions of the Association for Computational Linguistics, vol. 9, pp. 329–345, 2021. https://doi.org/10.1162/tacl_a_00369
Action Editor: Jimmy Lin. Submission batch: 6/2020; Revision batch: 9/2020; Published 4/2021.

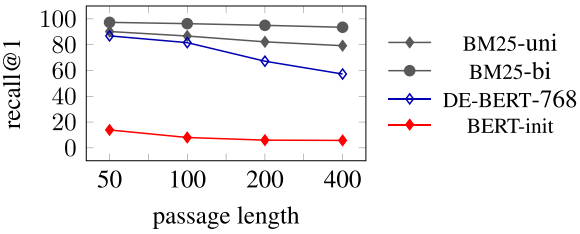


Figure 1: Recall@1 for retrieving passage containing a query from three million candidates. The figure compares a fine-tuned BERT-based dual encoder (DE-BERT-768), an off-the-shelf BERT-based encoder with average pooling (BERT-init), and sparse term-based retrieval (BM25), while binning passages by length.

encoders, and the circumstances in which these powerful models do not yet reach the state of the art. Here we explore these questions using both theoretical and empirical tools, and propose a new architecture that leverages the strengths of dual encoders while avoiding some of their weaknesses.

We begin with a theoretical investigation of compressive dual encoders—dense encodings whose dimension is below the vocabulary size—and analyze their ability to preserve distinctions made by sparse bag-of-words retrieval models, which we term their **fidelity**. Fidelity is important for the sub-problem of detecting precise term overlap, and is a tractable proxy for capacity. Using the theory of dimensionality reduction, we relate fidelity to the normalized margin between the gold retrieval result and its competitors, and show that this margin is in turn related to the length of documents in the collection. We validate the theory with an empirical investigation of the effects of random projection compression on sparse BM25 retrieval using queries and documents from TREC-CAR, a recent IR benchmark (Dietz et al., 2018).

Next, we offer a multi-vector encoding model, which is computationally feasible for retrieval like the dual-encoder architecture and achieves significantly better quality. A simple hybrid that interpolates models based on dense and sparse representations leads to further improvements.

We compare the performance of dual encoders, multi-vector encoders, and their sparse-dense hybrids with classical sparse retrieval models and attentional neural networks, as well as state-of-the-art published results where available. Our evaluations include open retrieval benchmarks (MS MARCO passage and document), and passage retrieval for question answering (Natural Questions). We confirm prior findings that full attentional architectures excel at reranking tasks, but are not efficient enough for large-scale retrieval. Of the more efficient alternatives, the hybridized multi-vector encoder is at or near the top in every evaluation, outperforming state-of-the-art retrieval results in MS MARCO. Our code is publicly available at `https://github.com/google-research/language/tree/master/language/multivec`.

2 Analyzing Dual Encoder Fidelity

A query or a document is a sequence of words drawn from some vocabulary $\mathcal{V}$. Throughout this section we assume a representation of queries and documents typically used in sparse bag-of-words models: Each query q and document d is a vector in $\mathbb{R}^v$ where v is the vocabulary size. We take the inner product $\langle q, d \rangle$ to be the relevance score of document d for query q. This framework accounts for a several well-known ranking models, including Boolean inner product, TF-IDF, and BM25.

We will compare sparse retrieval models with compressive dual encoders, for which we write $f(d)$ and $f(q)$ to indicate compression of d and q to $\mathbb{R}^k$, with $k \ll v$, and where k does not vary with the document length. For these models, the relevance score is the inner product $\langle f(q), f(d) \rangle$. (In §3, we consider encoders that apply to sequences of tokens rather than vectors of counts.)

A fundamental question is how the capacity of dual encoders varies with the embedding size k. In this section we focus on the related, more tractable notion of fidelity: How much can we compress the input while maintaining the ability to mimic the performance of bag-of-words retrieval? We explore this question mainly through the encoding model of random projections, but also discuss more general dimensionality reduction in §2.2.

2.1 Random Projections

To establish baselines on the fidelity of compressive dual encoder retrieval, we now consider encoders based on **random projections** (Vempala, 2004). The encoder is defined as $f(x) = Ax$, where $A \in \mathbb{R}^{k \times v}$ is a random matrix. In **Rademacher embeddings**, each element $a_{i,j}$

of the matrix A is sampled with equal probability from two possible values: $\{-\frac{1}{\sqrt{k}}, \frac{1}{\sqrt{k}}\}$. In **Gaussian embeddings**, each $a_{i,j} \sim N(0, k^{-1/2})$. A pairwise ranking error occurs when $\langle q, d_1\rangle > \langle q, d_2\rangle$ but $\langle Aq, Ad_1\rangle < \langle Aq, Ad_2\rangle$. Using such random projections, it is possible to bound the probability of any such pairwise error in terms of the embedding size.

Definition 2.1. *For a query q and pair of documents (d_1, d_2) such that $\langle q, d_1\rangle \geq \langle q, d_2\rangle$, the **normalized margin** is defined as, $\mu(q, d_1, d_2) = \frac{\langle q, d_1 - d_2\rangle}{\|q\| \times \|d_1 - d_2\|}$.*

Lemma 1. *Define a matrix $A \in \mathbb{R}^{k\times d}$ of Gaussian or Rademacher embeddings. Define vectors q, d_1, d_2 such that $\mu(q, d_1, d_2) = \epsilon > 0$. A ranking error occurs when $\langle Aq, Ad_2\rangle \geq \langle Aq, Ad_1\rangle$. If β is the probability of such an error then,*

$$\beta \leq 4\exp\left(-\frac{k}{2}(\epsilon^2/2 - \epsilon^3/3)\right). \quad (1)$$

The proof, which builds on well-known results about random projections, is found in §A.1. By solving (1) for k, we can derive an embedding size that guarantees a desired upper bound on the pairwise error probability,

$$k \geq 2(\epsilon^2/2 - \epsilon^3/3)^{-1}\ln\frac{4}{\beta}. \quad (2)$$

It is convenient to derive a simpler but looser quadratic bound (proved in §A.2):

Corollary 1. *Define vectors q, d_1, d_2 such that $\epsilon = \mu(q, d_1, d_2) > 0$. If $A \in \mathbb{R}^{k\times v}$ is a matrix of random Gaussian or Rademacher embeddings such that $k > 12\epsilon^{-2}ln\frac{4}{\beta}$, then $Pr(\langle Aq, Ad_1\rangle \leq \langle Aq, Ad_2\rangle) \leq \beta$.*

On the Tightness of the Bound. Let $k^*(q, d_1, d_2)$ denote the lowest dimension Gaussian or Rademacher random projection following the definition in Lemma 1, for which $\Pr(\langle Aq, Ad_1\rangle < \langle Aq, Ad_2\rangle) \leq \beta$, for a given document pair (d_1, d_2) and query q with normalized margin ϵ. Our lemma places an upper bound on k^*, saying that $k^*(q, d_1, d_2) \leq 2(\epsilon^2/2 - \epsilon^3/3)^{-1}\ln\frac{4}{\beta}$. Any $k \geq k^*(q, d_1, d_2)$ has sufficiently low probability of error, but lower values of k could potentially also have the desired property. Later in this section we perform empirical evaluation to study the tightness of the bound; although theoretical tightness (up to a constant factor) is suggested by results on the optimality of the distributional Johnson-Lindenstrauss lemma (Johnson and Lindenstrauss, 1984; Jayram and Woodruff, 2013; Kane et al., 2011), here we study the question only empirically.

2.1.1 Recall-at-*r*

In retrieval applications, it is important to return the desired result within the top r search results. For query q, define d_1 as the document that maximizes some inner product ranking metric. The probability of returning d_1 in the top r results after random projection can be bounded by a function of the embedding size and normalized margin:

Lemma 2. *Consider a query q, with target document d_1, and document collection $\mathcal{D}$ that excludes d_1, and such that $\forall d_2 \in \mathcal{D}, \mu(q, d_1, d_2) > 0$. Define r_0 to be any integer such that $1 \leq r_0 \leq |\mathcal{D}|$. Define ϵ to be the r_0'th smallest normalized margin $\mu(q, d_1, d_2)$ for any $d_2 \in \mathcal{D}$, and for simplicity assume that only a single document $d_2 \in \mathcal{D}$ has $\mu(q, d_1, d_2) = \epsilon$.*[2]

Define a matrix $A \in \mathbb{R}^{k\times d}$ of Gaussian or Rademacher embeddings. Define R to be a random variable such that $R = |\{d_2 \in \mathcal{D} : \langle Aq, Ad_1\rangle \leq \langle Aq, Ad_2\rangle\}|$, and let $C = 4(|\mathcal{D}| - r_0 + 1)$. Then

$$\Pr(R \geq r_0) \leq C\exp\left(-\frac{k}{2}(\epsilon^2/2 - \epsilon^3/3)\right).$$

The proof is in §A.3. A direct consequence of the lemma is that to achieve recall-at-$r_0 = 1$ for a given $(q, d_1, \mathcal{D})$ triple with probability $\geq 1 - \beta$, it is sufficient to set

$$k \geq \frac{2}{\epsilon^2/2 - \epsilon^3/3}\ln\frac{4(|\mathcal{D}| - r_0 + 1)}{\beta}, \quad (3)$$

where ϵ is the r_0'th smallest normalized margin.

As with the bound on pairwise relevance errors in Lemma 1, Lemma 2 implies an upper bound on the minimum random projection dimension $k^*(q, d_1, \mathcal{D})$ that recalls d_1 in the top r_0 results with probability $\geq 1 - \beta$. Due to the application of the union bound and worst-case assumptions about the normalized margins of documents in $\mathcal{D}_\epsilon$,

[2]The case where multiple documents are tied with normalized margin ϵ is straightforward but slightly complicates the analysis.

this bound is potentially loose. Later in this section we examine the empirical relationship between maximum document length, the distribution of normalized margins, and k^*.

2.1.2 Application to Boolean Inner Product

Boolean inner product is a retrieval function in which $d, q \in \{0,1\}^v$ over a vocabulary of size v, with d_i indicating the presence of term i in the document (and analogously for q_i). The relevance score $\langle q, d \rangle$ is then the number of terms that appear in both q and d. For this simple retrieval function, it is possible to compute an embedding size that guarantees a desired pairwise error probability over an entire dataset of documents.

Corollary 2. *For a set of documents $\mathcal{D} = \{d \in \{0,1\}^v\}$ and a query $q \in \{0,1\}^v$, let $L_D = \max_{d \in \mathcal{D}} \|d\|^2$ and $L_Q = \|q\|^2$. Let $A \in \mathbb{R}^{k \times v}$ be a matrix of random Rademacher or Gaussian embeddings such that $k \geq 24 L_Q L_D ln\frac{4}{\beta}$. Then for any $d_1, d_2 \in \mathcal{D}$ such that $\langle q, d_1 \rangle > \langle q, d_2 \rangle$, the probability that $\langle Aq, Ad_1 \rangle \leq \langle Aq, Ad_2 \rangle$ is $\leq \beta$.*

The proof is in §A.4. The corollary shows that for Boolean inner product ranking, we can guarantee any desired error bound β by choosing an embedding size k that grows linearly in L_D, the number of unique terms in the longest document.

2.1.3 Application to TF-IDF and BM25

Both TF-IDF (Spärck Jones, 1972) and BM25 (Robertson et al., 2009) can be written as inner products between bag-of-words representations of the document and query as described earlier in this section. Set the query representation $\tilde{q}_i = q_i \times \text{IDF}_i$, where q_i indicates the presence of the term in the query and IDF_i indicates the inverse document frequency of term i. The TF-IDF score is then $\langle \tilde{q}, d \rangle$. For BM25, we define $\tilde{d} \in \mathbb{R}^v$, with each $\tilde{d}_i$ a function of the count d_i and the document length (and hyperparameters); $\text{BM25}(q, d)$ is then $\langle \tilde{q}, \tilde{d} \rangle$. Due to its practical utility in retrieval, we now focus on BM25.

Pairwise Accuracy. We use empirical data to test the applicability of Lemma 1 to the BM25 relevance model. We select query-document triples (q, d_1, d_2) from the TREC-CAR dataset (Dietz et al., 2018) by considering all possible (q, d_2), and selecting $d_1 = \arg\max_d \text{BM25}(q, d)$. We bin the triples by the normalized margin ϵ, and compute the quantity $(\epsilon^2/2 - \epsilon^3/3)^{-1}$. According to Lemma 1, the minimum embedding size of a random projection k^* which has $\leq \beta$ probability of making an error on a triple with normalized margin ϵ is upper bounded by a linear function of this quantity. In particular, for $\beta = .05$, the Lemma entails that $k^* \leq 8.76(\epsilon^2/2 - \epsilon^3/3)^{-1}$. In this experiment we measure the empirical value of k^* to evaluate the tightness of the bound.

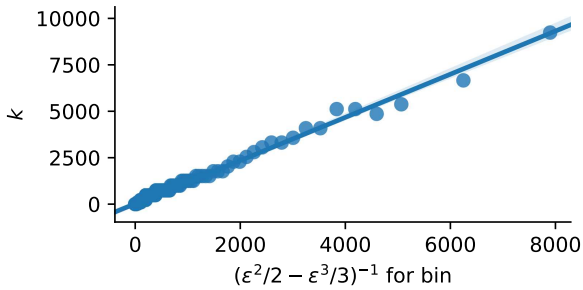
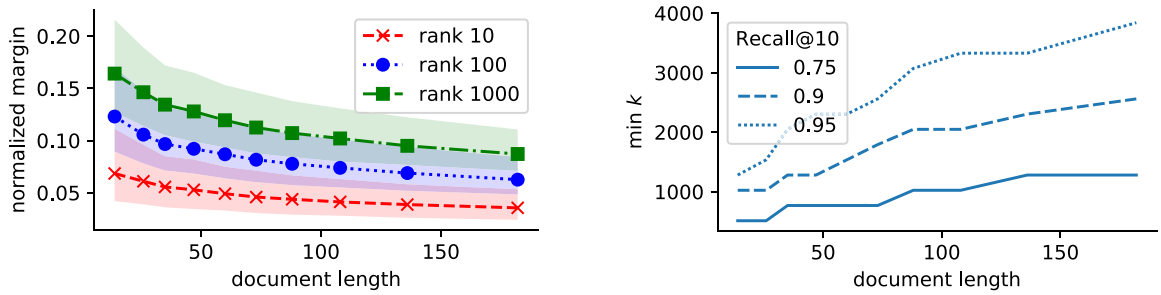


Figure 2: Minimum k sufficient for Rademacher embeddings to approximate BM25 pairwise rankings on TREC-CAR with error rate $\beta < .05$.

The results are shown on the x-axis of Figure 2. For each bin we compute the minimum embedding size required to obtain 95% pairwise accuracy in ranking d_1 vs d_2, using a grid of 40 possible values for k between 32 and 9472, shown on the y-axis. (We exclude examples that had higher values of $(\epsilon^2/2 - \epsilon^3/3)^{-1}$ than the range shown because they did not reach 95% accuracy for the explored range of k.) The figure shows that the theoretical bound is tight up to a constant factor, and that the minimum embedding size that yields desired fidelity grows linearly with $(\epsilon^2/2 - \epsilon^3/3)^{-1}$.

Margins and Document Length. For boolean inner product, it was possible to express the minimum possible normalized margin (and therefore a sufficient embedding size) in terms of L_Q and L_D, the maximum number of unique terms across all queries and documents, respectively. Unfortunately, it is difficult to analytically derive a minimum normalized margin ϵ for either TF-IDF or BM25: Because each term may have a unique inverse document frequency, the minimum non-zero margin $\langle q, d_1 - d_2 \rangle$ decreases with the number of terms in the query as each additional term creates more ways in which two documents can receive nearly the same score. We therefore study empirically how normalized margins vary with maximum document length. Using the TREC-CAR retrieval dataset, we bin documents by length. For each query, we compute the normalized margins between



(a) Each datapoint is the median normalized margin per bin, and the shaded areas show the 25th and 75th quantiles.

(b) Each line shows the minimum random projection dimension k that achieves a desired value of recall-at-10 for each bin of documents.

Figure 3: Random projection on BM25 retrieval in TREC-CAR dataset, with documents binned by length.

the document with best BM25 in the bin and all other documents in the bin, and look at the 10th, 100th, and 1000th smallest normalized margins. The distribution over these normalized margins is shown in Figure 3a, revealing that normalized margins decrease with document length. In practice, the observed minimum normalized margin for a collection of documents and queries is found to be much lower for BM25 compared to Boolean inner product. For example, for the collection used in Figure 2, the minimum normalized margin for BM25 is 6.8e-06, while for Boolean inner product it is 0.0169.

Document Length and Encoding Dimension. Figure 3b shows the growth in minimum random projection dimension required to reach desired recall-at-10, using the same document bins as in Figure 3a. As predicted, the required dimension increases with the document length, while the normalized margin shrinks.

2.2 Bounds on General Encoding Functions

We derived upper bounds on minimum required encoding for random linear projections above, and found the bounds on (q, d_1, d_2) triples to be empirically tight up to a constant factor. More general non-linear and learned encoders could be more efficient. However, there are general theoretical results showing that it is impossible for any encoder to guarantee an inner product distortion $|\langle f(x), f(y)\rangle - \langle x, y\rangle| \leq \epsilon$ with an encoding that does not grow as $\Omega(\epsilon^{-2})$ (Larsen and Nelson, 2017; Alon and Klartag, 2017), for vectors x, y with norm ≤ 1. These results suggest more general capacity limitations for fixed-length dual encoders when document length grows.

In our setting, BM25, TF-IDF, and Boolean inner product can all be reformulated equivalently as inner products in a space with vectors of norm at most 1 by L_2-normalizing each query vector and rescaling all document vectors by $\sqrt{L_D} = \max_d ||d||$, a constant factor that grows with the length of the longest document. Now suppose we desire to limit the distortion on the *unnormalized* inner products to some value $\leq \tilde{\epsilon}$, which might guarantee a desired performance characteristic. This corresponds to decreasing the maximum *normalized* inner product distortion ϵ by a factor of $\sqrt{L_D}$. According to the general bounds on dimensionality reduction mentioned in the previous paragraph, this could necessitate an increase in the encoding size by a factor of L_D.

However, there are a number of caveats to this theoretical argument. First, the theory states only that there *exist* vector sets that cannot be encoded into representations that grow more slowly than $\Omega(\epsilon^{-2})$; actual documents and queries might be easier to encode if, for example, they are generated from some simple underlying stochastic process. Second, our construction achieves $||d|| \leq 1$ by rescaling all document vectors by a constant factor, but there may be other ways to constrain the norms while using the embedding space more efficiently. Third, in the non-linear case it might be possible to eliminate ranking errors without achieving low inner product distortion. Finally, from a practical perspective, the generalization offered by learned dual encoders might overwhelm any sacrifices in fidelity, when evaluated on real tasks of

interest. Lacking theoretical tools to settle these questions, we present a set of empirical investigations in later sections of this paper. But first we explore a lightweight modification to the dual encoder, which offers gains in expressivity at limited additional computational cost.

3 Multi-Vector Encodings

The theoretical analysis suggests that fixed-length vector representations of documents may in general need to be large for long documents, if fidelity with respect to sparse high-dimensional representations is important. Cross-attentional architectures can achieve higher fidelity, but are impractical for large-scale retrieval (Nogueira et al., 2019b; Reimers and Gurevych, 2019; Humeau et al., 2020). We therefore propose a new architecture that represents each document as a fixed-size set of m vectors. Relevance scores are computed as the maximum inner product over this set.

Formally, let $\mathrm{x} = (x_1, \ldots, x_T)$ represent a sequence of tokens, with x_1 equal to the special token [CLS], and define y analogously. Then $[h_1(\mathrm{x}), \ldots, h_T(\mathrm{x})]$ represents the sequence of contextualized embeddings at the top level of a deep transformer. We define a single-vector representation of the query x as $f^{(1)}(\mathrm{x}) = h_1(\mathrm{x})$, and a multi-vector representation of document y as $f^{(m)}(\mathrm{y}) = [h_1(\mathrm{y}), \ldots, h_m(\mathrm{y})]$, the first m representation vectors for the sequence of tokens in y, with $m < T$. The relevance score is defined as $\max_{j=1\ldots m} \langle f^{(1)}(\mathrm{x}), f_j^{(m)}(\mathrm{y}) \rangle$.

Although this scoring function is not a dual encoder, the search for the highest-scoring document can be implemented efficiently with standard approximate nearest-neighbor search by adding multiple (m) entries for each document to the search index data structure. If some vector $f_j^{(m)}(\mathrm{y})$ yields the largest inner product with the query vector $f^{(1)}(\mathrm{x})$, it is easy to show the corresponding document must be the one that maximizes the relevance score $\psi^{(m)}(\mathrm{x}, \mathrm{y})$. The size of the index must grow by a factor of m, but due to the efficiency of contemporary approximate nearest neighbor and maximum inner product search, the time complexity can be sublinear in the size of the index (Andoni et al., 2019; Guo et al., 2016b). Thus, a model using m vectors of size k to represent documents is more efficient at run-time than a dual encoder that uses a single vector of size mk.

This efficiency is a key difference from the POLY-ENCODER (Humeau et al., 2020), which computes a fixed number of vectors per *query*, and aggregates them by softmax attention against document vectors. (Yang et al., 2018b) propose a similar architecture for language modeling. Because of the use of softmax in these approaches, it is not possible to decompose the relevance score into a max over inner products, and so fast nearest-neighbor search cannot be applied. In addition, these works did not address retrieval from a large document collection.

Analysis. To see why multi-vector encodings can enable smaller encodings per vector, consider an idealized setting in which each document vector is the sum of m orthogonal **segments** such that $d = \sum_{i=1}^{m} d^{(i)}$ and each query refers to exactly one segment in the gold document.[3] An orthogonal segmentation can be obtained by choosing the segments as a partition of the vocabulary.

Theorem 1. *Define vectors $q, d_1, d_2 \in \mathbb{R}^v$ such that $\langle q, d_1 \rangle > \langle q, d_2 \rangle$, and assume that both d_1 and d_2 can be decomposed into m segments such that: $d_1 = \sum_{i=1}^{m} d_1^{(i)}$, and analogously for d_2; all segments across both documents are orthogonal. If there exists an i such that $\langle q, d_1 \rangle = \langle q, d_1^{(i)} \rangle$ and $\langle q, d_2 \rangle \geq \langle q, d_2^{(i)} \rangle$, then $\mu(q, d_1^{(i)}, d_2^{(i)}) \geq \mu(q, d_1, d_2)$. (The proof is in §A.5.)*

Remark 1. *The BM25 score can be computed from non-negative representations of the document and query; if the segmentation corresponds to a partition of the vocabulary, then the segments will also be non-negative, and thus the condition $\langle q, d_2 \rangle \geq \langle q, d_2^{(i)} \rangle$ holds for all i.*

The relevant case is when the same segment is maximal for both documents, $\langle q, d_2^{(i)} \rangle = \max_j \langle q, d_2^{(j)} \rangle$, as will hold for ''simple'' queries that are well-aligned with the segmentation. Then the normalized margin in the multi-vector model will be at least as large as in the equivalent single vector representation. The relationship to encoding size follows from the theory in the previous section: Theorem 1 implies that if we set $f_i^{(m)}((\mathrm{y}) = Ad^{(i)}$ (for appropriate A), then

[3] Here we use (d, q) rather than (x, y) because we describe vector encodings rather than token sequences.

an increase in the normalized margin enables the use of a smaller encoding dimension k while still supporting the same pairwise error rate. There are now m times more ''documents'' to evaluate, but Lemma 2 shows that this exerts only a logarithmic increase on the encoding size for a desired recall@r. But while we hope this argument is illuminating, the assumptions of orthogonal segments and perfect segment match against the query are quite strong. We must therefore rely on empirical analysis to validate the efficacy of multi-vector encoding in realistic applications.

Cross-Attention. Cross-attentional architectures can be viewed as a generalization of the multi-vector model: (1) set $m = T_{\text{max}}$ (one vector per token); (2) compute one vector per token in the query; (3) allow more expressive aggregation over vectors than the simple $\max$ employed above. Any sparse scoring function (e.g., BM25) can be mimicked by a cross-attention model, which need only compute identity between individual words; this can be achieved by random projection word embeddings whose dimension is proportional to the log of the vocabulary size. By definition, the required representation also grows linearly with the number of tokens in the passage and query. As with the POLY-ENCODER, retrieval in the cross-attention model cannot be performed efficiently at scale using fast nearest-neighbor search. In contemporaneous work, Khattab and Zaharia (2020) propose an approach with T_Y vectors per query and T_X vectors per document, using a simple sum-of-max for aggregation of the inner products. They apply this approach to retrieval via re-ranking results of T_Y nearest-neighbor searches. Our multi-vector model uses fixed length representations instead, and a single nearest neighbor search per query.

4 Experimental Setup

The full IR task requires detection of both precise word overlap and semantic generalization. Our theoretical results focus on the first aspect, and derive theoretical and empirical bounds on the sufficient dimensionality to achieve high fidelity with respect to sparse bag-of-words models as document length grows, for two types of linear random projections. The theoretical setup differs from modeling for realistic information-seeking scenarios in at least two ways.

First, trained non-linear dual encoders might be able to detect precise word overlap with much lower-dimensional encodings, especially for queries and documents with a natural distribution, which may exhibit a low-dimensional subspace structure. Second, the semantic generalization aspect of the IR task may be more important than the first aspect for practical applications, and our theory does not make predictions about how encoder dimensionality relates to such ability to compute general semantic similarity.

We relate the theoretical analysis to text retrieval in practice through experimental studies on three tasks. The first task, described in §5, tests the ability of models to retrieve natural language documents that exactly contain a query and evaluates both BM25 and deep neural dual encoders on a task of detecting precise word overlap, defined over texts with a natural distribution. The second task, described in §6, is the passage retrieval sub-problem of the open-domain QA version of the Natural Questions (Kwiatkowski et al., 2019; Lee et al., 2019); this benchmark reflects the need to capture graded notions of similarly and has a natural query text distribution. For both of these tasks, we perform controlled experiments varying the maximum length of the documents in the collection, which enables assessing the relationship between encoder dimension and document length.

To evaluate the performance of our best models in comparison to state-of-the-art works on large-scale retrieval and ranking, in §7 we report results on a third group of tasks focusing on passage/document ranking: the passage and document-level MS MARCO retrieval datasets (Nguyen et al., 2016; Craswell et al., 2020). Here we follow the standard two-stage retrieval and ranking system: a first-stage retrieval from a large document collection, followed by reranking with a cross-attention model. We focus on the impact of the first-stage retrieval model.

4.1 Models

Our experiments compare compressive and sparse dual encoders, cross attention, and hybrid models.

BM25. We use case-insensitive wordpiece tokenizations of texts and default BM25 parameters from the *gensim* library. We apply either unigram (BM25-uni) or combined unigram+bigram representations (BM25-bi).

Dual Encoders from BERT (DE-BERT). We encode queries and documents using BERT-base, which is a pre-trained transformer network (12 layers, 768 dimensions) (Devlin et al., 2019). We implement dual encoders from BERT as a special case of the multi-vector model formalized in §3, with number of vectors for the document $m = 1$: The representations for queries and documents are the top layer representations at the [CLS] token. This approach is widely used for retrieval (Lee et al., 2019; Reimers and Gurevych, 2019; Humeau et al., 2020; Xiong et al., 2020).[4] For lower-dimensional encodings, we learn down-projections from $d = 768$ to $k \in 32, 64, 128, 512$,[5] implemented as a single feed-forward layer, followed by layer normalization. All parameters are fine-tuned for the retrieval tasks. We refer to these models as DE-BERT-k.

Cross-Attentional BERT. The most expressive model we consider is cross-attentional BERT, which we implement by applying the BERT encoder to the concatenation of the query and document, with a special [SEP] separator between x and y. The relevance score is a learned linear function of the encoding of the [CLS] token. Due to the computational cost, cross-attentional BERT is applied only in reranking as in prior work (Nogueira and Cho, 2019; Yang et al., 2019). These models are referred to as CROSS-ATTENTION.

Multi-Vector Encoding from BERT (ME-BERT). In §3 we introduced a model in which every document is represented by exactly m vectors. We use $m = 8$ as a good compromise between cost and accuracy in §5 and §6, and find values of 3 to 4 for m more accurate on the datasets in §7. In addition to using BERT output representations directly, we consider down-projected representations, implemented using a feed-forward layer with dimension $768 \times k$. A model with k-dimensional embeddings is referred to as ME-BERT-k.

Sparse-Dense Hybrids (HYBRID). A natural approach to balancing between the fidelity of sparse representations and the generalization of learned dense ones is to build a hybrid. To do this, we linearly combine a sparse and dense system's scores using a single trainable weight λ, tuned on a development set. For example, a hybrid model of ME-BERT and BM25-uni is referred to as HYBRID-ME-BERT-uni. We implement approximate search to retrieve using a linear combination of two systems by re-ranking n-best top scoring candidates from each system. Prior and concurrent work has also used hybrid sparse-dense models (Guo et al., 2016a; Seo et al., 2019; Karpukhin et al., 2020; Ma et al., 2020; Gao et al., 2020). Our contribution is to assess the impact of sparse-dense hybrids as the document length grows.

[4]Based on preliminary experiments with pooling strategies we use the [CLS] vectors (without the feed-forward projection learned on the next sentence prediction task).

[5]We experimented with adding a similar layer for $d = 768$, but this did not offer empirical gains.

4.2 Learning and Inference

For the experiments in §5 and §6, all trained models are initialized from BERT-base, and all parameters are fine-tuned using a cross-entropy loss with 7 sampled negatives from a pre-computed 200-document list and additional in-batch negatives (with a total number of 1024 candidates in a batch); the pre-computed candidates include 100 top neighbors from BM25 and 100 random samples. This is similar to the method by Lee et al. (2019), but with additional fixed candidates, also used in concurrent work (Karpukhin et al., 2020). Given a model trained in this way, for the scalable methods, we also applied hard-negative mining as in Gillick et al. (2019) and used one iteration when beneficial. More sophisticated negative selection is proposed in concurrent work (Xiong et al., 2020). For retrieval from large document collections with the scalable models, we used ScaNN: an efficient approximate nearest neighbor search library (Guo et al., 2020); in most experiments, we use exact search settings but also evaluate approximate search in Section §7. In §7, the same general approach with slightly different hyperparameters (detailed in that section) was used, to enable more direct comparisons to prior work.

5 Containing Passage ICT Task

We begin with experiments on the task of retrieving a Wikipedia passage y containing a sequence of words x. We create a dataset using Wikipedia, following the Inverse Cloze Task definition by Lee et al. (2019), but adapted to suit the goals of our study. The task is defined by first breaking Wikipedia texts into segments of length at most l.

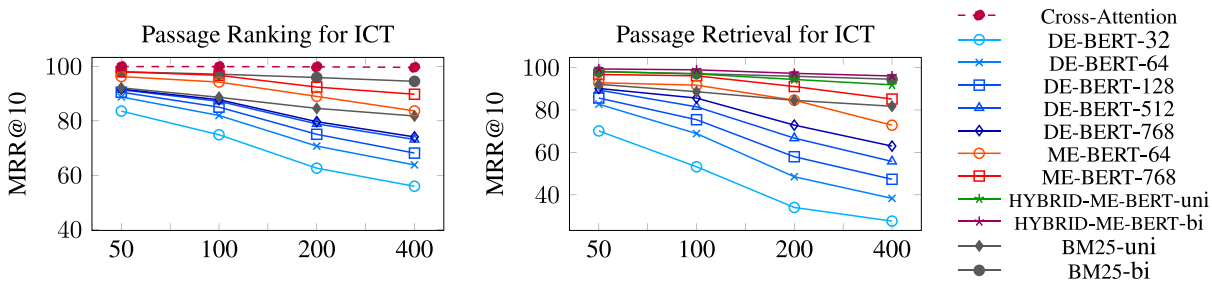


Figure 4: Results on the containing passage ICT task as maximum passage length varies (50 to 400 tokens). *Left*: Reranking 200 candidates; *Right*: Retrieval from 3 million candidates. Exact numbers refer to Table A.1.

These form the document collection $\mathcal{D}$. Queries x_i are generated by sampling sub-sequences from the documents y_i. We use queries of lengths between 5 and 25, and do not remove the queries x_i from their corresponding documents y_i.

We create a dataset with 1 million queries and evaluate retrieval against four document collections $\mathcal{D}_l$, for $l \in$ 50,100,200,400. Each $\mathcal{D}_l$ contains 3 million documents of maximum length l tokens. In addition to original Wikipedia passages, each $\mathcal{D}_l$ contains synthetic distractor documents, which contain the large majority of words in x but differ by one or two tokens. 5 K queries are used for evaluation, leaving the rest for training and validation. Although checking containment is a straightforward machine learning task, it is a good testbed for assessing the fidelity of compressive neural models. BM25-bi achieves over 95 MRR@10 across collections for this task.

Figure 4 (*left*) shows test set results on reranking, where models need to select one of 200 passages (top 100 BM25-bi and 100 random candidates). It is interesting to see how strong the sparse models are relative to even a 768-dimensional DE-BERT. As the document length increases, the performance of both the sparse and dense dual encoders worsens; the accuracy of the DE-BERT models falls most rapidly, widening the gap to BM25.

Full cross-attention is nearly perfect and does not degrade with document length. DE-BERT-768, which uses 8 vectors of dimension 768 to represent documents, strongly outperforms the best DE-BERT model. Even DE-BERT-64, which uses 8 vectors of size only 64 instead (thus requiring the same document collection size as DE-BERT-512 and being faster at inference time), outperforms the DE-BERT models by a large margin.

Figure 4 (*right*) shows results for the much more challenging task of retrieval from 3 million candidates. For the latter setting, we only evaluate models that can efficiently retrieve nearest neighbors from such a large set. We see similar behavior to the reranking setting, with the multi-vector methods exceeding BM25-uni performance for all lengths and DE-BERT models under-performing BM25-uni. The hybrid model outperforms both components in the combination with largest improvements over ME-BERT for the longest-document collection.

6 Retrieval for Open-Domain QA

For this task we similarly use English Wikipedia[6] as four different document collections, of maximum passage length $l \in \{50, 100, 200, 400\}$, and corresponding approximate sizes of 39 million, 27.3 million, 16.1 million, and 10.2 million documents, respectively. Here we use real user queries contained in the Natural Questions dataset (Kwiatkowski et al., 2019). We follow the setup in Lee et al. (2019). There are 87,925 QA pairs in training and 3,610 QA pairs in the test set. We hold out a subset of training for development.

For document retrieval, a passage is correct for a query x if it contains a string that matches exactly an annotator-provided short answer for the question. We form a reranking task by considering the top 100 results from BM25-uni and 100 random samples, and also consider the full retrieval setting. BM25-uni is used here instead of BM25-bi, because it is the stronger model for this task.

Our theoretical results do not make direct predictions for performance of compressive dual encoder models relative to BM25 on this task. They

[6] https://archive.org/download/enwiki-20181220.

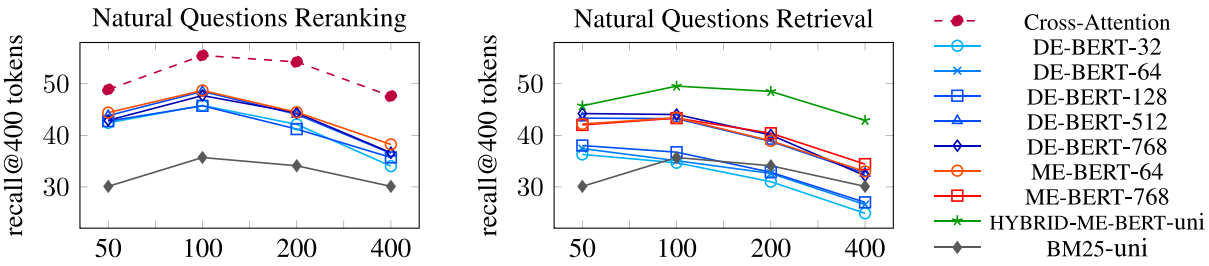


Figure 5: Results on NQ passage recall as maximum passage length varies (50 to 400 tokens). *Left*: Reranking of 200 passages; *Right*: Open domain retrieval result on all of (English) Wikipedia. Exact numbers refer to Table A.1.

do tell us that as the document length grows, low-dimensional compressive dual encoders may not be able to measure weighted term overlap precisely, potentially leading to lower performance on the task. Therefore, we would expect that higher dimensional dual encoders, multi-vector encoders, and hybrid models become more useful for collections with longer documents.

Figure 5 (*left*) shows heldout set results on the reranking task. To fairly compare systems that operate over collections of different-sized passages, we allow each model to select approximately the same number of tokens (400) and evaluate on whether an answer is contained in them. For example, models retrieving from $\mathcal{D}_{50}$ return their top 8 passages, and ones retrieving from $\mathcal{D}_{100}$ retrieve top 4. The figure shows this recall@400 tokens across models. The relative performance of BM25-uni and DE-BERT is different from that seen in the ICT task, due to the semantic generalizations needed. Nevertheless, higher-dimensional DE-BERT models generally perform better, and multi-vector models provide further benefits, especially for longer-document collections; ME-BERT-768 outperforms DE-BERT-768 and ME-BERT-64 outperforms DE-BERT-512; CROSS-ATTENTION is still substantially stronger.

Figure 5 (*right*) shows heldout set results for the task of retrieving from Wikipedia for each of the four document collections $\mathcal{D}_l$. Unlike the reranking setting, only higher-dimensional DE-BERT models outperform BM25 for passages longer than 50. The hybrid models offer large improvements over their components, capturing both precise word overlap and semantic similarity. The gain from adding BM25 to ME-BERT and DE-BERT increases as the length of the documents in the collection grows, which is consistent with our expectations based on the theory.

7 Large-Scale Supervised IR

The previous experimental sections focused on understanding the relationship between compressive encoder representation dimensionality and document length. Here we evaluate whether our newly proposed multi-vector retrieval model ME-BERT, its corresponding dual encoder baseline DE-BERT, and sparse-dense hybrids compare favorably to state-of-the-art models for large-scale supervised retrieval and ranking on IR benchmarks.

Datasets. The MS MARCO passage ranking task focuses on ranking passages from a collection of about 8.8 mln. About 532k queries paired with relevant passages are provided for training. The MS MARCO document ranking task is on ranking full documents instead. The full collection contains about 3 million documents and the training set has about 367 thousand queries. We report results on the passage and document development sets, comprising 6,980 and 5,193 queries, respectively in Table 1. We report MS MARCO and TREC DL 2019 (Craswell et al., 2020) test results in Table 2.

Model Settings. For MS MARCO passage we apply models on the provided passage collections. For MS MARCO document, we follow Yan et al. (2020) and break documents into a set of overlapping passages with length up to 482 tokens, each including the document URL and title. For each task, we train the models on that task's training data only. We initialize the retriever and reranker models with BERT-large. We train dense retrieval models on positive and negative candidates from the 1000-best list of BM25, additionally using one iteration of hard negative mining when beneficial. For ME-BERT, we used $m = 3$ for the passage and $m = 4$ for the document task.

		MS-Passage	MS-Doc
	Model	MRR	MRR
Retrieval	BM25	0.167	0.249
	BM25-E	0.184	0.209
	DOC2QUERY	0.215	-
	DOCT5QUERY	0.278	-
	DEEPCT	0.243	-
	HDCT	-	0.300
	DE-BERT	0.302	0.288
	ME-BERT	0.334	0.333
	DE-HYBRID	0.304	0.313
	DE-HYBRID-E	0.309	0.315
	ME-HYBRID	0.338	**0.346**
	ME-HYBRID-E	**0.343**	0.339
Reranking	MULTI-STAGE	0.390	-
	IDST	0.408	-
	Leaderboard	**0.439**	-
	DE-BERT	0.391	0.339
	ME-BERT	0.395	**0.353**
	ME-HYBRID	0.394	**0.353**

Table 1: Development set results on MS MARCO-Passage (MS-Passage), MS MARCO-Document (MS-Doc) showing MRR@10.

Model	MRR(MS)	RR	NDCG@10	Holes@10
Passage Retrieval				
BM25-Anserini	0.186	0.825	0.506	**0.000**
DE-BERT	0.295	0.936	0.639	0.165
ME-BERT	0.323	0.968	0.687	0.109
DE-HYBRID-E	0.306	0.951	0.659	0.105
ME-HYBRID-E	**0.336**	**0.977**	**0.706**	0.051
Document Retrieval				
Base-Indri	0.192	0.785	0.517	**0.002**
DE-BERT	-	0.841	0.510	0.188
ME-BERT	-	0.877	0.588	0.109
DE-HYBRID-E	0.287	0.890	0.595	0.084
ME-HYBRID-E	0.310	**0.914**	**0.610**	0.063

Table 2: Test set first-pass retrieval results on the passage and document TREC 2019 DL evaluation as well as MS MARCO eval MRR@10 (passage) and MRR@100 (document) under MRR(MS).

Results. Table 1 comparatively evaluates our models on the dev sets of two tasks. The state of the art prior work follows the two-stage retrieval and reranking approach, where an efficient first-stage system retrieves a (usually large) list of candidates from the document collection, and a second stage more expensive model such as cross-attention BERT reranks the candidates.

Our focus is on improving the first stage, and we compare to prior works in two settings: **Retrieval**, top part of Table 1, where only first-stage efficient retrieval systems are used and **Reranking**, bottom part of the table, where more expensive second-stage models are employed to re-rank candidates. Figure 6 delves into the impact of the first-stage retrieval systems as the number of candidates the second stage reranker has access to is substantially reduced, improving efficiency.

We report results in comparison to the following systems: 1) MULTI-STAGE (Nogueira and Lin, 2019), which reranks BM25 candidates with a cascade of BERT models, 2) DOC2QUERY (Nogueira et al., 2019b) and DOCT5QUERY (Nogueira and Lin, 2019), which use neural models to expand documents before indexing and scoring with sparse retrieval models, 3) DEEPCT (Dai and Callan, 2020b), which learns to map BERT's contextualized text representations to context-aware term weights, 4) HDCT (Dai and Callan, 2020a), which uses a hierachical approach that combines passage-level term weights into document level term weights, 5) IDST, a two-stage cascade ranking pipeline by Yan et al. (2020), and 6) Leaderboard, which is the best score on the MS MARCO-passage leaderboard as of Sept. 18, 2020.[7]

We also compare our models both to our own BM25 implementation described in §4.1, and the external publicly available sparse model implementations, denoted with BM25-E. For the passage task, BM25-E is the Anserini (Yang et al., 2018a) system with default parameters. For the document task, BM25-E is the official IndriQueryLikelihood baseline. We report on dense-sparse hybrids using both our own BM25, and the external sparse systems; the latter hybrids are indicated by a suffix -E.

Looking at the top part of Table 1, we can see that our DE-BERT model already outperforms or is competitive with prior systems. The multi-vector model brings larger improvement on the dataset containing longer documents (MS MARCO document), and the sparse-dense hybrid models bring improvements over dense-only models on both datasets. According to a Wilcoxon signed rank test for statistical significance, all differences between DE-BERT, ME-BERT, DE-HYBRID-E, and ME-HYBRID-E are statistically significant on both development sets with p-value $< .0001$.

When a large number of candidates can be reranked, the impact of the first-stage system decreases. In the bottom part of the table we

[7] https://microsoft.github.io/msmarco/.

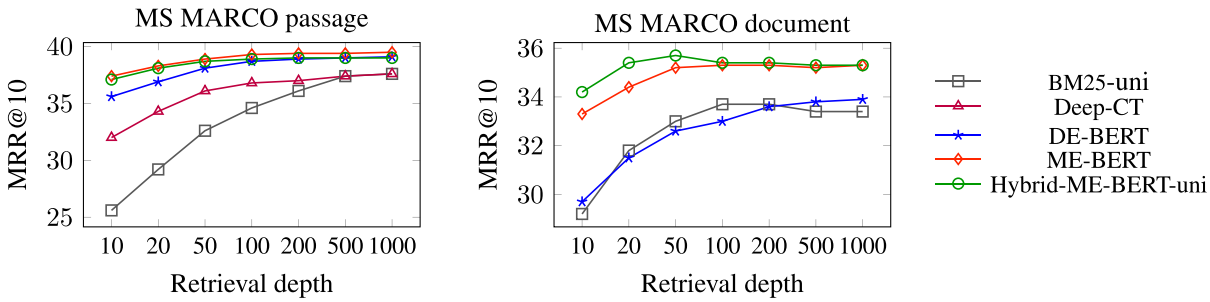


Figure 6: MRR@10 when reranking at different retrieval depth (10 to 1000 candidates) for MS MARCO.

see that our models are comparable to systems reranking BM25 candidates. The accuracy of the first-stage system is particularly important when the cost of reranking a large set of candidates is prohibitive. Figure 6 shows the performance of systems that rerank a smaller number of candidates. We see that, when a very small number of candidates can be scored with expensive cross-attention models, the multi-vector ME-BERT and hybrid models achieve large improvements compared to prior systems on both MS MARCO tasks.

Table 2 shows test results for dense models, external sparse model baselines, and hybrids of the two (without reranking). In addition to test set (eval) results on the MS MARCO passage task, we report metrics on the manually annotated passage and document retrieval test set at TREC DL 2019. We report the fraction of unrated items as Holes@10 following Xiong et al. (2020).

Time and Space Analysis Figure 7 compares the running time/quality trade-off curves for DE-BERT and ME-BERT on the MS MARCO passage task using the ScaNN (Guo et al., 2020) library on a 160 Intel(R) Xeon(R) CPU @ 2.20GHz cores machine with 1.88TB memory. Both models use one vector of size $k = 1024$ per query; DE-BERT uses one and ME-BERT uses 3 vectors of size $k = 1024$ per document. The size of the document index for DE-BERT is 34.2GB and the size of the index for ME-BERT is about 3 times larger. The indexing time was 1.52h and 3.02h for DE-BERT and ME-BERT, respectively. The ScaNN configuration we use is num_leaves=5000, and num_leaves_to_search ranges from 25 to 2000 (from less to more exact search) and time per query is measured when using parallel inference on all 160 cores. In the higher quality range of the curves, ME-BERT achieves

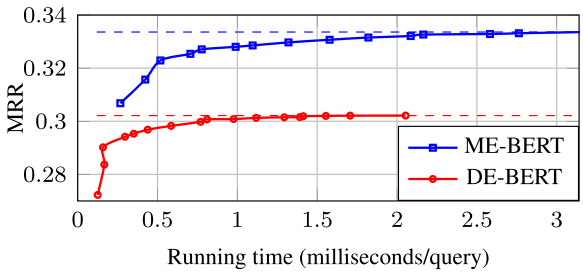


Figure 7: Quality/running time tradeoff for DE-BERT and ME-BERT on the MS MARCO passage dev set. Dashed lines show quality with exact search.

substantially higher MRR than DE-BERT for the same inference time per query.

8 Related Work

We have mentioned research on improving the accuracy of retrieval models throughout the paper. Here we focus on work related to our central focus on the capacity of dense dual encoder representations relative to sparse bags-of-words.

In compressive sensing it is possible to recover a bag of words vector x from the projection Ax for suitable A. Bounds for the sufficient dimensionality of isotropic Gaussian projections (Candes and Tao, 2005; Arora et al., 2018) are more pessimistic than the bound described in §2, but this is unsurprising because the task of recovering bags-of-words from a compressed measurement is strictly harder than recovering inner products.

Subramani et al. (2019) ask whether it is possible to exactly recover sentences (token sequences) from pretrained decoders, using vector embeddings that are added as a bias to the decoder hidden state. Because their decoding model is more expressive (and thus more computationally intensive) than inner product retrieval, the theoretical

issues examined here do not apply. Nonetheless, (Subramani et al., 2019) empirically observe a similar dependence between sentence length and embedding size. Wieting and Kiela (2019) represent sentences as bags of random projections, finding that high-dimensional projections ($k = 4096$) perform nearly as well as trained encoding models. These empirical results provide further empirical support for the hypothesis that bag-of-words vectors from real text are ''hard to embed'' in the sense of Larsen and Nelson (2017). Our contribution is to systematically explore the relationship between document length and encoding dimension, focusing on the case of exact inner product-based retrieval. We leave the combination of representation learning and approximate retrieval for future work.

9 Conclusion

Transformers perform well on an unreasonable range of problems in natural language processing. Yet the computational demands of large-scale retrieval push us to seek other architectures: cross-attention over contextualized embeddings is too slow, but dual encoding into fixed-length vectors may be insufficiently expressive, sometimes failing even to match the performance of sparse bag-of-words competitors. We have used both theoretical and empirical techniques to characterize the fidelity of fixed-length dual encoders, focusing on the role of document length. Based on these observations, we propose hybrid models that yield strong performance while maintaining scalability.

Acknowledgments

We thank Ming-Wei Chang, Jon Clark, William Cohen, Kelvin Guu, Sanjiv Kumar, Kenton Lee, Jimmy Lin, Ankur Parikh, Ice Pasupat, Iulia Turc, William A. Woods, Vincent Zhao, and the anonymous reviewers for helpful discussions of this work.

References

Dimitris Achlioptas. 2003. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *Journal of Computer and System Sciences*, 66(4):671–687. **DOI:** https://doi.org/10.1016/S0022-0000(03)00025-4

Noga Alon and Bo'az Klartag. 2017. Optimal compression of approximate inner products and dimension reduction. In *58th Annual Symposium on Foundations of Computer Science (FOCS)*. **DOI:** https://doi.org/10.1109/FOCS.2017.65

Alexandr Andoni, Piotr Indyk, and Ilya Razenshteyn. 2019. Approximate nearest neighbor search in high dimensions. *Proceedings of the International Congress of Mathematicians (ICM 2018)*.

Sanjeev Arora, Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. 2018. A compressed sensing view of unsupervised text embeddings, bag-of-n-grams, and LSTMs. In *Proceedings of the International Conference on Learning Representations (ICLR)*. **DOI:** https://doi.org/10.1142/9789813272880_0182

Shai Ben-David, Nadav Eiron, and Hans Ulrich Simon. 2002. Limitations of learning via embeddings in Euclidean half spaces. *Journal of Machine Learning Research*, 3(Nov):441–461.

Emmanuel J. Candes and Terence Tao. 2005. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12): 4203–4215. **DOI:** https://doi.org/10.1109/TIT.2005.858979

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. In *Text REtrieval Conference (TREC)*. TREC.

Zhuyun Dai and Jamie Callan. 2020a. Context-aware document term weighting for ad-hoc search. In *Proceedings of The Web Conference 2020*. **DOI:** https://doi.org/10.1145/3366423.3380258

Zhuyun Dai and Jamie Callan. 2020b. Context-aware sentence/passage term importance estimation for first stage retrieval. *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*.

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*.

Laura Dietz, Ben Gamari, Jeff Dalton, and Nick Craswell. 2018. TREC complex answer retrieval overview. In *Text REtrieval Conference (TREC)*.

Luyu Gao, Zhuyun Dai, Zhen Fan, and Jamie Callan. 2020. Complementing lexical retrieval with semantic residual embedding. *CoRR*, abs/2004.13969. Version 1.

Daniel Gillick, Sayali Kulkarni, Larry Lansing, Alessandro Presta, Jason Baldridge, Eugene Ie, and Diego Garcia-Olano. 2019. Learning dense representations for entity retrieval. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. **DOI:** https://doi.org/10.18653/v1/K19-1049

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016a. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*.

Ruiqi Guo, Sanjiv Kumar, Krzysztof Choromanski, and David Simcha. 2016b. Quantization based fast inner product search. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. 2020. Accelerating large-scale inference with anisotropic vector quantization. In *Proceedings of the 37th International Conference on Machine Learning*.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the International Conference on Information and Knowledge Management (CIKM)*. **DOI:** https://doi.org/10.1145/2505515.2505665

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Thathachar S. Jayram and David P. Woodruff. 2013. Optimal bounds for Johnson-Lindenstrauss transforms and streaming problems with subconstant error. *ACM Transactions on Algorithms (TALG)*, 9(3):1–17. **DOI:** https://doi.org/10.1145/2483699.2483706

William B. Johnson and Joram Lindenstrauss. 1984. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary Mathematics*, 26(189–206):1.

Daniel Kane, Raghu Meka, and Jelani Nelson. 2011. Almost optimal explicit Johnson-Lindenstrauss families. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*.

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)*. **DOI:** https://doi.org/10.18653/v1/2020.emnlp-main.550

Omar Khattab and Matei Zaharia. 2020. ColBERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. **DOI:** https://doi.org/10.1145/3397271.3401075

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*, 7:453–466. **DOI:**

https://doi.org/10.1162/tacl_a_00276

Cody Kwok, Oren Etzioni, and Daniel S. Weld. 2001. Scaling question answering to the web. *ACM Transactions on Information Systems (TOIS)*, 19(3):242–262. **DOI:** https://doi.org/10.1145/502115.502117

Kasper Green Larsen and Jelani Nelson. 2017. Optimality of the Johnson-Lindenstrauss lemma. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. **DOI:** https://doi.org/10.1109/FOCS.2017.64

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the Association for Computational Linguistics (ACL)*.

Ji Ma, Ivan Korotkov, Yinfei Yang, Keith Hall, and Ryan T. McDonald. 2020. Zero-shot neural retrieval via domain-targeted synthetic query generation. *CoRR*, abs/2004.14503.

Bhaskar Mitra and Nick Craswell. 2018. An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval*, 13(1):1–126. **DOI:** https://doi.org/10.1561/1500000061

Karthik Narasimhan, Adam Yala, and Regina Barzilay. 2016. Improving information extraction by acquiring external evidence with reinforcement learning. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*. **DOI:** https://doi.org/10.18653/v1/D16-1261

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.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with BERT. *CoRR*, abs/1901.04085.

Rodrigo Nogueira and Jimmy Lin. 2019. From doc2query to doctttttquery. https://github.com/castorini/docTTTTTquery

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019a. Multi-stage document ranking with BERT. *CoRR*, abs/1910.14424.

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

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*. **DOI:** https://doi.org/10.18653/v1/D19-1410

Stephen Robertson, and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389. **DOI:** https://doi.org/10.1561/1500000019

Minjoon Seo, Jinhyuk Lee, Tom Kwiatkowski, Ankur Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2019. Real-time open-domain question answering with dense-sparse phrase index. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Karen Spärck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21. **DOI:** https://doi.org/10.1108/eb026526

Nishant Subramani, Samuel Bowman, and Kyunghyun Cho. 2019. Can unconditional language models recover arbitrary sentences? In *Advances in Neural Information Processing Systems*.

Santosh S. Vempala. 2004. *The Random Projection Method*, volume 65. American Mathematical Society.

Ellen M. Voorhees. 2001. The TREC question answering track. *Natural Language Engineering*, 7(4):361–378. **DOI:** https://doi.org/10.1017/S1351324901002789

John Wieting and Douwe Kiela. 2019. No training required: Exploring random encoders for sentence classification. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *CoRR*, abs/2007.00808. Version 1.

Ming Yan, Chenliang Li, Chen Wu, Bin Bi, Wei Wang, Jiangnan Xia, and Luo Si. 2020. IDST at TREC 2019 deep learning track: Deep cascade ranking with generation-based document expansion and pre-trained language modeling. In *Text REtrieval Conference (TREC)*.

Peilin Yang, Hui Fang, and Jimmy Lin, New York, NY, USA. 2018a. Anserini: Reproducible ranking baselines using lucene. *Journal of Data and Information Quality*, 10(4). **DOI:** https://doi.org/10.1145/3239571

Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Simple applications of BERT for ad hoc document retrieval. *CoRR*, abs/1903.10972.

Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W. Cohen. 2018b. Breaking the softmax bottleneck: A high-rank RNN language model. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

A Proofs

A.1 Lemma 1

Proof. For both distributions of embeddings, the error on the squared norm can be bounded with high probability (Achlioptas, 2003, Lemma 5.1):

$$\begin{aligned}&\Pr(\left|\|Ax\|^2 - \|x\|^2\right| > \epsilon\|x\|^2) \\ &< 2\exp(-\frac{k}{2}(\epsilon^2/2 - \epsilon^3/3)).\end{aligned} \tag{4}$$

This bound implies an analogous bound on the absolute error of the inner product (Ben-David et al., 2002, corollary 19),

$$\begin{aligned}&\Pr(|\,\langle Ax, Ay\rangle - \langle x, y\rangle\,| \geq \frac{\epsilon}{2}(\|x\|^2 + \|y\|^2)) \\ &\leq 4\exp(-\frac{k}{2}(\epsilon^2/2 - \epsilon^3/3)).\end{aligned} \tag{5}$$

Let $\bar{q} = q/\|q\|$ and $\bar{d} = (d_1 - d_2)/\|d_1 - d_2\|$. Then $\mu(q, d_1, d_2) = \langle \bar{q}, \bar{d}\rangle$. A ranking error occurs if and only if $\langle A\bar{q}, A\bar{d}\rangle \leq 0$, which implies $|\,\langle A\bar{q}, A\bar{d}\rangle - \langle \bar{q}, \bar{d}\rangle\,| \geq \epsilon$. By construction $\|\bar{q}\| = \|\bar{d}\| = 1$, so the probability of an inner product distortion $\geq \epsilon$ is bounded by the right-hand side of (5). □

A.2 Corollary 1

Proof. We have $\epsilon = \mu(q, d_1, d_2) = \langle \bar{q}, \bar{d}\rangle \leq 1$ by the Cauchy-Schwarz inequality. For $\epsilon \leq 1$, we have $\epsilon^2/6 \leq \epsilon^2/2 - \epsilon^3/3$. We can then loosen the bound in (1) to $\beta \leq 4\exp(-\frac{k}{2}\frac{\epsilon^2}{6})$. Taking the natural log yields $\ln\beta \leq \ln 4 - \epsilon^2 k/12$, which can be rearranged into $k \geq 12\epsilon^{-2}\ln\frac{4}{\beta}$. □

A.3 Lemma 2

Proof. For convenience define $\mu(d_2) = \mu(q, d_1, d_2)$. Define ϵ as in the theorem statement, and $\mathcal{D}_\epsilon = \{d_2 \in \mathcal{D} : \mu(q, d_1, d_2) \geq \epsilon\}$. We have

$$\begin{aligned}&\Pr(R \geq r_0) \leq \Pr(\exists d_2 \in \mathcal{D}_\epsilon : Aq_1 \leq Aq_2) \\ &\leq \sum_{d_2 \in \mathcal{D}_\epsilon} 4\exp(-\frac{k}{2}(\mu(d_2)^2/2 - \mu(d_2)^3/3)) \\ &\leq 4|\mathcal{D}_\epsilon|\exp(-\frac{k}{2}(\epsilon^2/2 - \epsilon^3/3)).\end{aligned}$$

The first inequality follows because the event $R \geq r_0$ implies the event $\exists d_2 \in \mathcal{D}_\epsilon : Aq_1 \leq Aq_2$. The second inequality follows by a combination of Lemma 1 and the union bound. The final inequality follows because for any $d_2 \in \mathcal{D}_\epsilon$, $\mu(q, d_1, d_2) \geq \epsilon$. The theorem follows because $|\mathcal{D}_\epsilon| = |\mathcal{D}| - r_0 + 1$. □

A.4 Corollary 2

Proof. For the retrieval function $\max_d\langle q, d\rangle$, the minimum non-zero unnormalized margin $\langle q, d_1\rangle - \langle q, d_2\rangle$ is 1 when q and d are Boolean vectors. Therefore the normalized margin has lower bound $\mu(q, d_1, d_2) \geq 1/(\|q\| \times \|d_1 - d_2\|)$. For non-negative d_1 and d_2 we have $\|d_1 - d_2\| \leq \sqrt{\|d_1\|^2 + \|d_2\|^2} \leq \sqrt{2L_D}$. Preserving a normalized margin of $\epsilon = (2L_Q L_D)^{-\frac{1}{2}}$ is therefore sufficient to avoid any pairwise errors. By plugging this value into Corollary 1, we see that setting $k \geq 24 L_Q L_D \ln\frac{4}{\beta}$ ensures that the probability of any pairwise error is $\leq \beta$. □

Model	Reranking				Retrieval			
Passage length	50	100	200	400	50	100	200	400
ICT task (MRR@10)								
CROSS-ATTENTION	99.9	99.9	99.8	99.6	-	-	-	-
HYBRID-ME-BERT-uni	-	-	-	-	98.2	97.0	94.4	91.9
HYBRID-ME-BERT-bi	-	-	-	-	99.3	99.0	97.3	96.1
ME-BERT-768	98.0	96.7	92.4	89.8	96.8	96.1	91.1	85.2
ME-BERT-64	96.3	94.2	89.0	83.7	92.9	91.7	84.6	72.8
DE-BERT-768	91.7	87.8	79.7	74.1	90.2	85.6	72.9	63.0
DE-BERT-512	91.4	87.2	78.9	73.1	89.4	81.5	66.8	55.8
DE-BERT-128	90.5	85.0	75.0	68.1	85.7	75.4	58.0	47.3
DE-BERT-64	88.8	82.0	70.7	63.8	82.8	68.9	48.5	38.3
DE-BERT-32	83.6	74.9	62.6	55.9	70.1	53.2	34.0	27.6
BM25-uni	92.1	88.6	84.6	81.8	92.1	88.6	84.6	81.8
BM25-bi	98.0	97.1	95.9	94.5	98.0	97.1	95.9	94.5
NQ (Recall@400 tokens)								
CROSS-ATTENTION	48.9	55.5	54.2	47.6	-	-	-	-
HYBRID-ME-BERT-uni	-	-	-	-	45.7	49.5	48.5	42.9
ME-BERT-768	43.6	49.6	46.5	38.7	42.0	43.3	40.4	34.4
ME-BERT-64	44.4	48.7	44.5	38.2	42.2	43.4	38.9	33.0
DE-BERT-768	42.9	47.7	44.4	36.6	44.2	44.0	40.1	32.2
DE-BERT-512	43.8	48.5	44.1	36.5	43.3	43.2	38.8	32.7
DE-BERT-128	42.8	45.7	41.2	35.7	38.0	36.7	32.8	27.0
DE-BERT-64	42.6	45.7	42.5	35.4	37.4	35.1	32.6	26.6
DE-BERT-32	42.4	45.8	42.1	34.0	36.3	34.7	31.0	24.9
BM25-uni	30.1	35.7	34.1	30.1	30.1	35.7	34.1	30.1

Table A.1: Results on ICT task and NQ task (correspond to Figure 4 and Figure 5).

A.5 Theorem 1

Proof. Recall that $\mu(q, d_1, d_2) = \frac{\langle q, d_1 - d_2 \rangle}{\|q\| \times \|d_1 - d_2\|}$. By assumption we have $\langle q, d_1^{(i)} \rangle = \langle q, d_1 \rangle$ and $\max_j \langle q, d_2^{(j)} \rangle \leq \langle q, d_2 \rangle$, implying that

$$\langle q, d_1^{(i)} - d_2^{(i)} \rangle \geq \langle q, d_1 - d_2 \rangle \tag{6}$$

In the denominator, we expand $\|d_1 - d_2\| = \|(d_1^{(i)} - d_2^{(i)}) + (d_1^{(\neg i)} - d_2^{(\neg i)})\|$, where $d^{(\neg i)} = \sum_{j \neq i} d^{(j)}$. Plugging this into the squared norm,

$$\|d_1 - d_2\|^2$$

$$= \|(d_1^{(i)} - d_2^{(i)}) + (d_1^{(\neg i)} - d_2^{(\neg i)})\|^2 \tag{7}$$

$$= \|d_1^{(i)} - d_2^{(i)}\|^2 + \|d_1^{(\neg i)} - d_2^{(\neg i)}\|^2 + 2\langle d_1^{(i)} - d_2^{(i)} \rangle, d_1^{(\neg i)} - d_2^{(\neg i)} \tag{8}$$

$$= \|d_1^{(i)} - d_2^{(i)}\|^2 + \|d_1^{(\neg i)} - d_2^{(\neg i)}\|^2 \tag{9}$$

$$\geq \|d_1^{(i)} - d_2^{(i)}\|^2. \tag{10}$$

The inner product $\langle d_1^{(i)} - d_2^{(i)}, d_1^{(\neg i)} - d_2^{(\neg i)} \rangle = 0$ because the segments are orthogonal. The combination of (6) and (10) completes the theorem. □