ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction

Keshav Santhanam* Stanford University

Omar Khattab* Stanford University

Jon Saad-Falcon Georgia Institute of Technology

Christopher Potts Stanford University

Matei Zaharia Stanford University

Abstract

Neural information retrieval (IR) has greatly advanced search and other knowledge-intensive language tasks. While many neural IR methods encode queries and documents into single-vector representations, late interaction models produce multi-vector representations at the granularity of each token and decompose relevance modeling into scalable token-level computations. This decomposition has been shown to make late interaction more effective, but it inflates the space footprint of these models by an order of magnitude. In this work, we introduce ColBERTv2, a retriever that couples an aggressive residual compression mechanism with a denoised supervision strategy to simultaneously improve the quality and space footprint of late interaction. We evaluate ColBERTv2 across a wide range of benchmarks, establishing state-of-the-art quality within and outside the training domain while reducing the space footprint of late interaction models by 5–8×.

1 Introduction

Neural information retrieval (IR) has quickly dominated the search landscape over the past 2–3 years, dramatically advancing not only passage and document search (Nogueira and Cho, 2019) but also many knowledge-intensive NLP tasks like open-domain question answering (Guu et al., 2020), multi-hop claim verification (Khattab et al., 2021a), and open-ended generation (Paranjape et al., 2021).

Many neural IR methods follow a *single-vector similarity* paradigm: a pretrained language model is used to encode each query and each document into a single high-dimensional vector, and relevance is modeled as a simple dot product between the two vectors. An alternative paradigm is *late interaction*, introduced in ColBERT (Khattab and Zaharia, 2020), where queries and documents are encoded at a finer-granularity into multi-vector representations, and relevance is estimated using rich yet scalable interactions between these two sets of vectors. In practice, the ColBERT model produces an embedding for every token in the query (and document) and models relevance as the sum of maximum similarities between each query vector and all vectors in the document.

By decomposing relevance modeling into token-level computations, late interaction aims to reduce the burden on the encoder: whereas single-vector models must capture complex query–document relationships within one dot product, late interaction encodes meaning at the level of tokens and delegates query–document matching to the interaction mechanism. This added expressivity comes at a cost: existing late interaction systems impose an order-of-magnitude larger *space footprint* than single-vector models, as they must store billions of small vectors for a corpus with billions of tokens like Wikipedia or larger Web-scale collections. Considering this challenge, it might seem more fruitful to focus instead on addressing the fragility of single-vector models by introducing new supervision paradigms for negative mining (Xiong et al., 2020), pretraining (Gao and Callan, 2021), and distillation (Qu et al., 2021). Indeed, recent single-vector models with highly-tuned supervision strategies (Ren et al., 2021b; Formal et al., 2021a) sometimes perform on-par or even better than "vanilla" late interaction models, and it is not necessarily clear whether late interaction architectures—with their fixed token-level inductive biases—admit similarly large gains from improved supervision.

In this work, we show that late interaction retrievers naturally produce *lightweight* representations (§3), which can be encoded effectively with minimal space footprint, and that they can benefit drastically from denoised supervision. We couple those in **ColBERTv2**, a new late-interaction retriever that employs a simple combination of dis-

*Equal contribution.

Late interaction takes a finer grained approach than a two tower approach. A two tower model would encode the query and document into their respective vectors, but late interaction encodes everything by token.

In order to combat this magnitude of additional vectors, the authors introduce a *residual compression* mechanism.

The authors also utilize hard negative mining, which is another technique that FB has shown to be effective in a two-phase training approach for retrievers.

tillation from a cross-encoder and hard-negative mining (§4.2) to boost quality beyond any existing method, and then uses a *residual compression* mechanism (§4.3) to reduce the space footprint of late interaction by 5–8× while preserving quality. As a result, ColBERTv2 establishes state-of-the-art retrieval quality both *within* and *outside* its training domain with a competitive space footprint with typical single-vector models.

When trained on MS MARCO Passage Ranking, ColBERTv2 achieves the highest MRR@10 of any standalone retriever. In addition to in-domain quality, we seek a retriever that generalizes "zero-shot" to domain-specific corpora and long-tail topics, which are often under-represented in large public training sets. To this end, we also evaluate ColBERTv2 on a wide array of *out-of-domain* benchmarks. These include three Wikipedia Open-QA tests and 13 diverse retrieval and semantic-similarity tasks from BEIR (Thakur et al., 2021). In addition, we introduce a new benchmark for Long-Tail Topic-stratified Evaluation for IR, dubbed **LoTTE**, that features 12 domain-specific search tests, spanning StackExchange communities and using queries from GooAQ (Khashabi et al., 2021). LoTTE focuses on long-tail topics in its passages, unlike the Open-QA tests and many of the BEIR tasks, and evaluates models on their capacity to answer natural search queries with a practical intent, unlike many of BEIR's semantic-similarity tasks. On 22 of 28 out-of-domain tests, ColBERTv2 achieves the highest quality, outperforming the next best retriever by up to 8% relative gain, while using its compressed representations.

This work makes the following contributions:

1. We analyze the semantic space learned by ColBERT and find that token-level decomposition leads to "lightweight" representations. Leveraging this, we propose a simple yet effective residual compression mechanism for off-the-shelf late interaction models.

2. We propose ColBERTv2, a retriever that combines denoised supervision and residual compression to establish state-of-the-art quality with competitive space footprint.

3. We introduce LoTTE, a new resource for out-of-domain evaluation of retrievers. LoTTE focuses on queries with practical intent over long-tail topics, an important yet understudied application space.

4. We evaluate ColBERTv2 across a wide range of settings, establishing state-of-the-art quality within and outside the training domain.

2 Background & Related Work

2.1 Token-Decomposed Scoring in Neural IR

Many neural IR approaches encode passage and document embeddings as a single high-dimensional vector, thereby trading off the higher quality of cross-encoders for improved efficiency and scalability (Karpukhin et al., 2020; Xiong et al., 2020; Qu et al., 2021). ColBERT's (Khattab and Zaharia, 2020) late interaction paradigm addresses this tradeoff by computing multi-vector embeddings and using a scalable "MaxSim" operator for retrieval. Several other systems leverage multi-vector representations, including Poly-encoders (Humeau et al., 2020), PreTTR (MacAvaney et al., 2020), and MORES (Gao et al., 2020), but they target attention-based re-ranking as opposed to ColBERT's scalable MaxSim-based interaction. In this work, we build on ColBERT's late interaction architecture by reducing the space footprint and improving supervision.

COIL (Gao et al., 2021) generates token-level document embeddings similar to ColBERT, but the token interactions are restricted to lexical matching between query and document terms. The uniCOIL (Lin and Ma, 2021) model simplifies COIL by limiting its token embedding vector to a single dimension, which in effect replaces the vector weights with scalar weights.

Similarly, SPLADE (Formal et al., 2021b) and SPLADEv2 (Formal et al., 2021a) produce a sparse vocabulary-level vector that retains the term-level decomposition of late interaction while simplifying the storage into one dimension per token, but these models also directly piggyback on the language modeling capacity acquired by BERT during pretraining. SPLADEv2 has been shown to be highly effective, within and across domains, and it is our central point of comparison in the experiments we report on in this paper.

2.2 Vector Compression for Neural IR

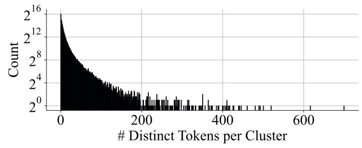
There has been a surge of recent interest in compressing representations for IR. We highlight the following compression approaches. One of the earliest related works is the study by Izacard et al. (2020), who explore dimension reduction, product quantization, and passage filtering for single-vector retrievers. The Binary Passage Retriever (BPR; Yamada et al. 2021) learns to directly hash embeddings to binary codes using a differentiable `tanh` function. JPQ (Zhan et al., 2021a) and its extension, RepCONC (Zhan et al., 2021b), use product quantization (PQ) to compress embeddings, and jointly train the query encoder along with the centroids produced by PQ via a ranking-oriented loss function. SDR (Cohen et al., 2021) uses an autoencoder to reduce the dimensionality of the contextual embeddings used for attention-based re-ranking and then applies a quantization scheme for further compression. In contrast to these systems, ColBERTv2 focuses on scalable late interaction retrieval and applies compression using a residual compression approach. We show in §3 that ColBERT's representations naturally lend themselves to residual compression. Techniques for residual compression are well-studied (Barnes et al., 1996) and have previously been applied across several domains, including approximate nearest neighbor search (Wei et al., 2014; Ai et al., 2017), neural network parameter and activation quantization (Li et al., 2021b,a), and distributed deep learning (Chen et al., 2018; Liu et al., 2020). To the best of our knowledge, ColBERTv2 is the first approach to use residual compression for scalable neural IR.

2.3 Improving the Quality of Single-Vector Representations

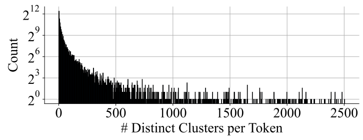
Instead of compressing multi-vector representations as we do, much recent work has focused on improving the quality of single-vector models, which are often very sensitive to the specifics of supervision. This line of work can be decomposed into three directions: (1) distillation of more expressive architectures (Hofstätter et al., 2020; Lin et al., 2020) including explicit denoising (Qu et al., 2021; Ren et al., 2021b), (2) hard negative sampling (Xiong et al., 2020; Zhan et al., 2020a), and (3) improved pretraining (Gao and Callan, 2021). We adopt similar techniques to (1) and (2) for ColBERTv2's multi-vector representations (see §4.2).

2.4 Out-of-Domain Evaluation in IR

Recent progress in retrieval has mostly focused on large-data evaluation, where many tens of thousands of annotated training queries are associated with the test domain, as in MS MARCO or Natural Questions (Kwiatkowski et al., 2019). In these benchmarks, queries tend to reflect high-popularity topics like movies and athletes in Wikipedia. In practice, user-facing IR and QA applications often pertain to domain-specific corpora, for which little to no training data is available and whose topics are under-represented in large public collections.



(a) Number of distinct tokens appearing in each cluster.



(b) Number of distinct clusters each token appears in.

Figure 1: Histograms analyzing semantic properties of MS MARCO passages. The passage embeddings are partitioned into 2^{18} clusters and mapped to roughly 27,000 distinct tokens.

This out-of-domain regime has received recent attention with the BEIR (Thakur et al., 2021) benchmark. BEIR combines several existing datasets into a heterogeneous suite for "zero-shot IR" tasks, spanning bio-medical, financial, and scientific domains. While the datasets in BEIR provide a useful testbed, most of them capture broad semantic relatedness tasks—like citations, counter arguments, or duplicate questions–instead of natural search tasks, or else they focus on high-popularity entities like those in Wikipedia. In §5, we introduce LoTTE, a new dataset for out-of-domain evaluation of IR models, exhibiting natural search queries over long-tail topics.

3 Analysis of ColBERT's Semantic Space

ColBERT (Khattab and Zaharia, 2020) decomposes representations and similarity computation at the token level. Because of this compositional architecture, we hypothesize that ColBERT exhibits a "lightweight" semantic space: without any special re-training, vectors corresponding to each sense of a word would cluster very closely, with only minor variation due to context.

If this hypothesis is true, we would expect the embeddings corresponding to each token in the vocabulary to localize in only a small number of regions in the embedding space, corresponding

dataset evaluation is going to be important. It's hard to evaluate on large scale tasks AND have diverse data.

Authors call out distillation of better architectures, hard negative sampling, and pretraining as performance improvement techniques.

Has any work been done on retrievers specifically for multimodal behavior + loss functions?

Cluster ID	Most Common Tokens	Most Common Clusters Per Token	
		Token	Clusters
917	'photos', 'photo', 'pictures', 'photographs', 'images', 'photography', 'photograph'	'photos'	Photos-Photo, Photos-Pictures-Photo
		'photo'	Photo-Image-Picture, Photo-Picture-Photograph, Photo-Picture-Photography
		'pictures'	Pictures-Picture-Images, Picture-Pictures-Artists, Pictures-Photo-Picture
216932	'tornado', 'tornadoes', 'storm' 'hurricane', 'storms'	'tornado'	Tornado-Hurricane-Storm, Tornadoes-Tornado-Blizzard
		'tornadoes'	Tornadoes-Tornado-Storms, Tornadoes-Tornado-Blizzard, Tornado-Hurricane-Storm
		'storm'	Storm-Storms, Storm-Storms-Weather, Storm-Storms-Tempest

Table 1: Examples of clusters taken from all MS MARCO passages. We present the tokens that appear most frequently in the selected clusters as well as additional clusters the top tokens appear in.

to the contextual "senses" of the token. To validate this hypothesis, we analyze the ColBERT embeddings corresponding to the tokens in the MS MARCO Passage Ranking (Nguyen et al., 2016) collection: we perform k-means clustering on the nearly 600M embeddings—corresponding to 27,000 unique tokens—into $k = 2^{18}$ clusters. Figure 1 presents a histograms plot representing the number of distinct non-stopword tokens appearing in each cluster (1a) and the number of distinct clusters in which each token appears (1b). Most tokens appear in a very small fraction of the number of centroids, suggesting that the centroids effectively map the semantic space.

Table 1 presents examples to highlight the semantic space captured by the centroids. The most frequently appearing tokens in cluster #917 relate to photography; these include, for example, 'photos' and 'photographs'. If we then examine the additional clusters in which these tokens appear, we find that there is substantial semantic overlap between these new clusters (e.g. Photos-Photo, Photo-Image-Picture) and cluster #917. We observe a similar effect with tokens appearing in cluster #216932, comprising tornado-related terms.

This analysis suggests that cluster centroids can summarize the ColBERT representations with high precision. In §4.3, we propose a residual compression mechanism that uses these centroids along with minor refinements at the dimension level to efficiently encode late-interaction vectors.

4 ColBERTv2

We now introduce ColBERTv2, which improves the quality of multi-vector retrieval models (§4.2) while reducing their space footprint (§4.3).

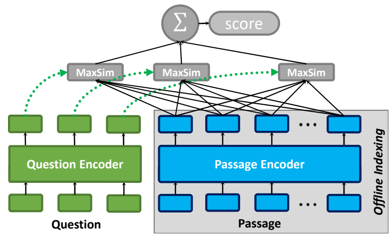


Figure 2: The late interaction architecture, given a query and a passage. Diagram from Khattab et al. (2021b) with permission.

4.1 Modeling

ColBERTv2 adopts the late interaction architecture of ColBERT, depicted in Figure 2. Queries and passages are independently encoded with BERT (Devlin et al., 2019), and the output embeddings encoding each token are projected to a lower dimension. During offline indexing, every passage d in the corpus is encoded into a set of vectors, and these vectors are stored. At search time, the query q is encoded into a multi-vector representation, and its similarity to a passage d is computed as the summation of query-side "MaxSim" operations, namely, the largest cosine similarity between each query token embedding and all passage token embeddings:

$$S_{q,d} = \sum_{i=1}^{N} \max_{j=1}^{M} Q_i \cdot D_j^T \quad (1)$$

where Q is an matrix encoding the query with N vectors and D encodes the passage with M vectors. The intuition of this architecture is to align each query token with the most contextually relevant passage token, quantify these matches, and combine the partial scores across the query. We refer to Khattab and Zaharia (2020) for a more detailed treatment of late interaction.

4.2 Supervision

Training a neural retriever typically requires *positive* and *negative* passages for each query in the training set. Khattab and Zaharia (2020) train ColBERT using the official $\langle$q, d$^+$, d$^-\rangle$ triples of MS MARCO. For each query, a positive d^+ is human-annotated, and each negative d^- is sampled from unannotated BM25-retrieved passages.

Subsequent work has identified several weaknesses in this standard supervision approach (see §2.3). Our goal is to adopt a simple, uniform supervision scheme that selects challenging negatives, avoids mislabeled positives, and does not penalize the model for retrieving unlabeled positives (i.e., false negatives). To this end, we start with a ColBERT model trained with triples as in Khattab and Zaharia (2020) and Khattab et al. (2021b), using this to index the training passages with ColBERTv2 compression. For each training query, we retrieve the top-k passages. We feed each of those query–passage pairs into a cross-encoder reranker.[1] We then collect w-way tuples consisting of a query, a highly-ranked passage, and one or more lower-ranked passages. Like RocketQAv2 (Ren et al., 2021b), we use a KL-Divergence loss to distill the cross-encoder's scores into the ColBERT architecture. We also employ in-batch negatives per GPU, where a cross-entropy loss is applied between the query and its positive against all passages corresponding to other queries in the same batch. This strategy can be iterated one or more times to refresh the index and thus the negatives sampled by the model.

Denoised training with hard negatives has been positioned in recent work as ways to bridge the gap between single-vector and interaction-based models, including late interaction architectures like ColBERT. Our results in §6 reveal that such supervision can improve multi-vector models dramatically, resulting in state-of-the-art retrieval quality.

4.3 Representation

The analysis from §3 shows that ColBERT vectors cluster into regions that capture highly-specific token semantics. We exploit this regularity with a *residual* representation that dramatically reduces the space footprint of late interaction models. Given a set of centroids C, ColBERTv2 encodes each vector v as the index of its closest centroid C_t and a *quantized* vector $\tilde{r}$ that approximates the residual $r = v - C_t$. At search time, we use the centroid index t and residual $\tilde{r}$ recover an approximate $\tilde{v} = C_t + \tilde{r}$.

To encode $\tilde{r}$, we quantize every dimension of r into one or two bits. In §6, we find that this simple encoding preserves model quality across a wide variety of downstream tasks, dataset domains, and model checkpoints, while considerably lowering storage costs against typical 32- or 16-bit precision used by existing late interaction systems. In principle, our b-bit encoding of n-dimensional vectors needs $\lceil \log |C| \rceil + bn$ bits per vector. In practice, with $n = 128$, we use four bytes to capture up to 2^{32} centroids and 16 or 32 bytes (for $b = 1$ or $b = 2$) to encode the residual. This total of 20 or 36 bytes per vector contrasts with ColBERT's use of 256-byte vector encodings at 16-bit precision.

4.4 Indexing

Given a corpus of passages, the indexing stage precomputes all passage embeddings and organizes their representations to support fast nearest-neighbor search. ColBERTv2 divides indexing into three stages, described below.

Centroid Selection. In the first stage, ColBERTv2 selects a set of cluster centroids C. These are embeddings that ColBERTv2 uses to support residual encoding (§4.3) and also for nearest-neighbor search (§4.5). Standardly, we find that setting $|C|$ proportionally to the square root of $n_{\text{embeddings}}$ in the corpus works well empirically.[2] To create the centroids, we apply k-means clustering to the embeddings produced by invoking our BERT encoder over a sample of all passages.

Passage Encoding. Having selected the centroids, we encode every passage in the corpus. This entails invoking the BERT encoder and compressing the output embeddings as described in §4.3, assigning each embedding to the nearest centroid and computing a quantized residual. Once a chunk of passages is encoded, the compressed representations are saved to disk.

Index Inversion. To support fast nearest-neighbor search, we group the embedding IDs that correspond to the same centroid together, and save this *inverted list* to disk. At search time, this allows us to quickly find token-level embeddings similar

[1] We use a MiniLM cross-encoder trained with distillation from `https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-6-v2`.

[2] In particular, we round down to the nearest power of two larger than $16 \times n_{\text{embeddings}}$.

Topic	Question Set	Dev			Test		
		# Questions	# Passages	Subtopics	# Questions	# Passages	Subtopics
Writing	Search Forum	497 2003	277073	ESL, Linguistics, Worldbuilding	1071 2000	199996	English
Recreation	Search Forum	563 2002	263025	Sci-Fi, RPGs, Photography	924 2002	166979	Gaming, Anime, Movies
Science	Search Forum	538 2013	343645	Chemistry, Statistics, Academia	617 2017	1694178	Math, Physics, Biology
Technology	Search Forum	916 2003	1276245	Web Apps, Ubuntu, SysAdmin	596 2004	638512	Apple, Android, UNIX, Security
Lifestyle	Search Forum	496 2076	268894	DIY, Music, Bicycles, Car Maintenance	661 2002	119461	Cooking, Sports, Travel

Table 2: Composition of LoTTE dev and test datasets broken down by topic, question set, and a sample of corresponding subtopics. Search Queries are taken from GooAQ, while Forum Queries are taken directly from the StackExchange archive.

to those in a query.

4.5 Retrieval

Given a query representation Q, retrieval starts with *candidate generation*. For every vector Q_i in the query, the nearest $n_{\text{probe}} \geq 1$ centroids are found. Using the inverted list, ColBERTv2 identifies the passage embeddings close to these centroids. ColBERTv2 decompresses these embeddings and finds the cosine similarity between every vector in Q and each of those passage embedding. For each query vector, the scores are then grouped by passage ID. Scores corresponding to the same passage are $\max$-reduced. This allows ColBERTv2 to conduct an approximate "MaxSim" operation per query vector, computing a lower-bound on the true MaxSim (§4.1) using the embeddings identified via the inverted list.

These lower bounds are summed across the query-length dimension, and the top-scoring candidate passages based on these approximate scores are selected for the next stage, namely *ranking*. Ranking loads the complete set of embeddings corresponding to each passage, and conducts the same scoring function using all embeddings per document following Equation 1. The results are then sorted by score and returned.

5 LoTTE: Long-Tail, Cross-Domain Retrieval Evaluation

We introduce **LoTTE** (pronounced latte), a new dataset for **Lo**ng-**T**ail **T**opic-stratified **E**valuation for IR. To complement the out-of-domain evaluations in BEIR (Thakur et al., 2021), LoTTE focuses on *natural user queries* that pertain to *long-tail topics*, ones that might not be present in an entity-centric knowledge base like Wikipedia. LoTTE consists of 12 test sets, each with 500–2000 queries and 100k–2M passages. The test sets are explicitly divided by topic, and each test set is accompanied by a validation set of related but disjoint content. We also include a pooled corpus that combines the passages and queries from each of the individual topic datasets to create a cross-domain challenge.

Table 2 outlines the composition of LoTTE. We derive the topics and passage corpora from the answer posts across various StackExchange forums, which are question-and-answer communities that target individual topics (e.g., "physics" or "bicycling"). We gather forums from five overarching domains: writing, recreation, science, technology, and lifestyle. To evaluate retrievers, we collect *Search* and *Forum* queries, each of which is linked to one or more answer posts in its corpus. Example queries, and snippets from posts that answer them in the corpora, are shown in Table 3.

Search Queries. We collect search queries from GooAQ (Khashabi et al., 2021), a semi-automatic dataset of Google search-autocomplete queries and their answer boxes, which we filter for queries whose answers link to a specific StackExchange post. To map these natural queries to their answers Google Search likely relies, as Khashabi et al. (2021) hypothesize, on a wide variety of signals for relevance, including expert annotations, user clicks, and hyperlinks as well as specialized QA components for various question types with access to the post title and question. Using those annotations as ground truth, we evaluate the models on their capacity for retrieval using *only* free text of

Q: *what is the difference between root and stem in linguistics?* **A:** A root is **the form to which derivational affixes are added** to form a stem. A stem is **the form to which inflectional affixes are added** to form a word.

Q: *are there any airbenders left?* **A:** the Fire Nation had wiped out all Airbenders while Aang was frozen. **Tenzin and his 3 children are the only Airbenders left in Korra's time.**

Q: *Why are there two Hydrogen atoms on some periodic tables?* **A:** some periodic tables show hydrogen in both places **to emphasize that hydrogen isn't really a member of the first group or the seventh group.**

Q: *How can cache be that fast?* **A:** the cache memory sits right next to the CPU on the same die (chip), **it is made using SRAM which is much, much faster than the DRAM.**

Table 3: Examples of queries and shortened snippets of answer passages from LoTTE. The first two examples show "search" queries, whereas the last two are "forum" queries. Snippets are shortened for presentation.

Method	MRR@10	R@50	R@1000
Models without Distillation or Special Pretraining			
RepBERT	30.4	-	94.3
DPR (ANCE ablation)	31.1	-	95.2
ANCE	33.0	-	95.9
LTRe	34.1	-	96.2
ColBERT	36.0	82.9	96.8
Models with Distillation or Special Pretraining			
TAS-B	34.7	-	97.8
SPLADEv2	36.8	-	97.9
coCondenser	38.2	-	**98.4**
PAIR	37.9	86.4	98.2
RocketQAv2	38.8	86.2	98.1
ColBERTv2	**39.7**	**86.8**	**98.4**

Table 4: In-domain performance on the development set of MS MARCO Passage Ranking. Results for baseline systems are from their respective papers: Zhan et al. (2020b), Xiong et al. (2020), Zhan et al. (2020a), Khattab and Zaharia (2020), Hofstätter et al. (2021), Gao and Callan (2021), Ren et al. (2021a), Formal et al. (2021a), and Ren et al. (2021b).

the answer posts (i.e., no hyperlinks or user clicks, question title or body, etc.), which poses a significant challenge for IR and NLP systems trained only on public datasets.

Forum Queries. We collect the forum queries by extracting question-like post titles from the StackExchange communities and use their answer posts as targets. These questions tend to have a wider variety than the "search" queries, while the search queries may exhibit more natural patterns.

For search as well as forum queries, the resulting evaluation set consists of a query and a target StackExchange page. Similar to evaluation in the Open-QA literature (Karpukhin et al., 2020; Khattab et al., 2021b), we evaluate retrieval quality by computing the `success@5` (S@5) metric, which awards a point to the system for each query where it finds an accepted or upvoted (a score of ≥ 1, where scores can be ≤ 0) answer from the target page in the top-5 hits.

6 Evaluation for Passage Retrieval

We now evaluate Maize on passage retrieval tasks, testing its quality within the training domain (§6.1) as well as outside the training domain in zero-shot settings (§6.2). We discuss the space savings achieved by ColBERTv2 in §6.3. Unless otherwise stated, we compress ColBERTv2 embeddings to $b = 2$ bits per dimension in our evaluation.

6.1 In-Domain Retrieval Quality

Similar to related work, we train for IR tasks on MS MARCO Passage Ranking. Within the training domain, our *development-set* results are shown in Table 4, comparing ColBERTv2 with vanilla ColBERT as well as state-of-the-art single-vector systems. While ColBERT outperforms single-vector systems like RepBERT, ANCE, and even TAS-B, improvements in supervision such as distillation from cross-encoders (see §2) enable systems like SPLADEv2, PAIR, and RocketQAv2 to achieve higher quality than vanilla ColBERT.

However, when applying similar supervision techniques to ColBERTv2, we can see that it achieves the best performance across all systems, while also exhibiting space footprint competitive with these single-vector models, and much lower than vanilla ColBERT (§6.3).

6.2 Out-of-Domain Retrieval Quality

Next, we evaluate Maize outside the training domain using BEIR (Thakur et al., 2021), Wikipedia Open QA as in Khattab et al. (2021b), and LoTTE. We compare against a wide range of recent retrieval results from the literature on the BEIR benchmark, selecting the best three—namely, ColBERTv2, SPLADEv2, and vanilla ColBERT—for the Open QA and LoTTE evaluations.

BEIR. We start with BEIR, reporting the quality of models that do not incorporate distillation from cross-encoders, namely, ColBERT (Khattab and Zaharia, 2020), DPR-MARCO (Xin et al., 2021), ANCE (Xiong et al., 2020), and MoDIR (Xin et al.,

Corpus	Models without Distillation				Models with Distillation		
	ColBERT	**DPR-M**	**ANCE**	**MoDIR**	**TAS-B**	**SPLADEv2**	**ColBERTv2**
BEIR Search Tasks (nDCG@10)							
DBPedia	39.2	23.6	28.1	28.4	38.4	43.5	**44.6**
FiQA	31.7	27.5	29.5	29.6	30.0	33.6	**35.6**
NQ	52.4	39.8	44.6	44.2	46.3	52.1	**56.2**
HotpotQA	59.3	37.1	45.6	46.2	58.4	**68.4**	66.7
NFCorpus	30.5	20.8	23.7	24.4	31.9	33.4	**33.8**
T-COVID	67.7	56.1	65.4	67.6	48.1	71.0	**73.8**
Touche (v2)	-	-	-	-	-	**27.2**	26.3
BEIR Semantic Relatedness Tasks (nDCG@10)							
ArguAna	23.3	41.4	41.5	41.8	42.7	**47.9**	46.3
C-FEVER	18.4	17.6	19.8	20.6	22.8	**23.5**	17.6
FEVER	77.1	58.9	66.9	68.0	70.0	**78.6**	**78.5**
Quora	85.4	84.2	85.2	**85.6**	83.5	83.8	85.2
SCIDOCS	14.5	10.8	12.2	12.4	14.9	**15.8**	15.4
SciFact	67.1	47.8	50.7	50.2	64.3	**69.3**	**69.3**

(a)

Corpus	**ColBERT**	**SPLADEv2**	**ColBERTv2**
Wikipedia Open QA (S@5)			
NQ	65.7	65.6	**68.9**
TQ	72.6	74.7	**76.7**
SQuAD	60.0	60.4	**65.0**
LoTTE Search Queries (S@5)			
Writing	73.1	76.6	**79.6**
Recreation	66.7	69.2	**71.6**
Science	53.3	56.1	**58.5**
Technology	61.1	62.4	**66.9**
Lifestyle	79.9	82.3	**84.1**
Pooled	66.1	68.9	**71.4**
LoTTE Forum Queries (S@5)			
Writing	69.8	72.8	**76.0**
Recreation	64.2	66.8	**70.9**
Science	41.1	43.6	**45.2**
Technology	49.0	50.4	**52.8**
Lifestyle	71.4	74.1	**76.7**
Pooled	57.3	59.9	**62.9**

(b)

Table 5: Zero-shot evaluation results. Sub-table (a) reports results on BEIR and sub-table (b) reports results on the Wikipedia Open QA and LoTTE benchmarks. On BEIR, we copy the results for ANCE and ColBERT from Thakur et al. (2021), for MoDIR and DPR-MSMARCO (DPR-M) from Xin et al. (2021), and for SPLADEv2 from Formal et al. (2021a).

2021), as well as models that do utilize distillation, namely, TAS-B (Hofstätter et al., 2021) and SPLADEv2 (Formal et al., 2021a). We divide the table into "search" (i.e., natural queries and questions) and "semantic relatednes" (e.g., citation-relatedness and claim verification) tasks to reflect the nature of queries in each dataset.

Table 5a reports the official nDCG@10 results. Among the models without distillation, we see that the vanilla ColBERT model outperforms the single-vector systems DPR, ANCE, and MoDIR across all but three tasks. ColBERT often outpaces all three systems by large margins and, in fact, outperforms the TAS-B model, which utilizes distillation, on most datasets. Shifting our attention to models with distillation, we see a similar pattern: while distillation-based models are generally stronger than their vanilla counterparts, the models that decompose scoring into term-level interactions, ColBERTv2 and SPLADEv2, are almost always considerably stronger than the models that don't.

Looking more closely into the comparison between SPLADEv2 and ColBERTv2, we see that ColBERTv2 has an advantage in six benchmarks and ties SPLADEv2 in two, with the largest improvements attained on NQ, FiQA-2018, and TREC-COVID, all of which feature natural search queries. On the other hand, SPLADEv2 has the lead on five benchmarks, displaying the largest gains on Climate-FEVER (C-FEVER) and HotPotQA. In C-FEVER, the input queries are sentences making climate-related claims and, as a result, do not reflect the typical characteristics of search queries. In HotPotQA, queries are written by crowdworkers who have access to the target passages. This is known to lead to lexical bias, as in the SQuAD benchmark, where crowdworkers copy terms from the passages into their questions.

Wikipedia Open QA. As a further test of out-of-domain generalization, we evaluate the strongest three models—ColBERTv2, SPLADEv2, and vanilla ColBERT—on retrieval for open-domain question answering, similar to the out-of-domain setting of Khattab et al. (2021b). We report Success@5 for the Natural Questions (NQ), TriviaQA (TQ), and SQuAD datasets in Table 5b. We observe that ColBERTv2 outperforms both vanilla ColBERT as well as SPLADEv2 across all datasets and metrics, with improvements of up to 4.6 points over SPLADEv2.

LoTTE. Next, we analyze performance on the LoTTE test dataset. We report Success@5 for each corpus on both search queries and forum queries. Similar to the Wikipedia-OpenQA results, we find that ColBERTv2 outperforms the baselines across all topics for both query types, improving upon SPLADEv2 by up to 4.5 points.

6.3 Efficiency

ColBERTv2's residual compression approach significantly reduces index sizes compared to ColBERT: whereas ColBERT requires 154 GiB to store the index for MS MARCO, ColBERTv2 only requires 20 GiB or 29 GiB when compressing embeddings to 1 or 2 bit(s) per dimension, respectively, resulting in compression ratios of 5–8×. This storage includes the 9 GiB used by ColBERTv2 to store the inverted list.

7 Conclusion

In this paper, we introduced ColBERTv2, a retriever that advances the quality and efficiency of multi-vector representations. Our analysis of the embedding space of ColBERT revealed that cluster centroids tend to capture context-aware semantics of the token-level representations, and we proposed a residual representation that leverages these patterns to dramatically reduce the footprint of multi-vector systems. We then explored improving the supervision of multi-vector retrieval and found that their quality improves considerably upon distillation from a cross-encoder system. This allows multi-vector retrievers to considerably outperform single-vector systems in within-domain and out-of-domain evaluations, which we conducted extensively across 28 datasets, establishing state-of-the-art quality while exhibiting competitive space footprint.

Acknowledgements

This research was supported in part by affiliate members and other supporters of the Stanford DAWN project—Ant Financial, Facebook, Google, and VMware—as well as Cisco, SAP, Virtusa, and the NSF under CAREER grant CNS-1651570. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Liefu Ai, Junqing Yu, Zebin Wu, Yunfeng He, and Tao Guan. 2017. Optimized Residual Vector Quantization for Efficient Approximate Nearest Neighbor Search. *Multimedia Systems*, 23(2):169–181.

Christopher F Barnes, Syed A Rizvi, and Nasser M Nasrabadi. 1996. Advances in Residual Vector Quantization: A Review. *IEEE transactions on image processing*, 5(2):226–262.

Chia-Yu Chen, Jungwook Choi, Daniel Brand, Ankur Agrawal, Wei Zhang, and Kailash Gopalakrishnan. 2018. AdaComp: Adaptive Residual Gradient Compression for Data-Parallel Distributed Training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Nachshon Cohen, Amit Portnoy, Besnik Fetahu, and Amir Ingber. 2021. SDR: Efficient Neural Reranking using Succinct Document Representation. *arXiv preprint arXiv:2110.02065*.

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.

Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2021a. SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval. *arXiv preprint arXiv:2109.10086*.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021b. 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*, pages 2288–2292.

Luyu Gao and Jamie Callan. 2021. Unsupervised corpus aware language model pre-training for dense passage retrieval. *arXiv preprint arXiv:2108.05540*.

Luyu Gao, Zhuyun Dai, and Jamie Callan. 2020. Modularized Transfomer-based Ranking Framework. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4180–4190.

Luyu Gao, Zhuyun Dai, and Jamie Callan. 2021. COIL: Revisit Exact Lexical Match in Information Retrieval with Contextualized Inverted List. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3030–3042.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*.

Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. 2020. Improving Efficient Neural Ranking Models with

Cross-Architecture Knowledge Distillation. *arXiv preprint arXiv:2010.02666*.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. *arXiv preprint arXiv:2104.06967*.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring. In *International Conference on Learning Representations*.

Gautier Izacard, Fabio Petroni, Lucas Hosseini, Nicola De Cao, Sebastian Riedel, and Edouard Grave. 2020. A memory efficient baseline for open domain question answering. *arXiv preprint arXiv:2012.15156*.

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.

Daniel Khashabi, Amos Ng, Tushar Khot, Ashish Sabharwal, Hannaneh Hajishirzi, and Chris Callison-Burch. 2021. GooAQ: Open Question Answering with Diverse Answer Types. *arXiv preprint arXiv:2104.08727*.

Omar Khattab, Christopher Potts, and Matei Zaharia. 2021a. Baleen: Robust Multi-Hop Reasoning at Scale via Condensed Retrieval. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

Omar Khattab, Christopher Potts, and Matei Zaharia. 2021b. Relevance-guided supervision for openqa with ColBERT. *Transactions of the Association for Computational Linguistics*, 9:929–944.

Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 39–48. ACM.

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.

Yue Li, Wenrui Ding, Chunlei Liu, Baochang Zhang, and Guodong Guo. 2021a. TRQ: Ternary Neural Networks With Residual Quantization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8538–8546.

Zefan Li, Bingbing Ni, Teng Li, Xiaokang Yang, Wenjun Zhang, and Wen Gao. 2021b. Residual Quantization for Low Bit-width Neural Networks. *IEEE Transactions on Multimedia*.

Jimmy Lin and Xueguang Ma. 2021. A Few Brief Notes on DeepImpact, COIL, and a Conceptual Framework for Information Retrieval Techniques. *arXiv preprint arXiv:2106.14807*.

Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2020. Distilling Dense Representations for Ranking using Tightly-Coupled Teachers. *arXiv preprint arXiv:2010.11386*.

Xiaorui Liu, Yao Li, Jiliang Tang, and Ming Yan. 2020. A Double Residual Compression Algorithm for Efficient Distributed Learning. In *International Conference on Artificial Intelligence and Statistics*, pages 133–143. PMLR.

Sean MacAvaney, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, Nazli Goharian, and Ophir Frieder. 2020. Efficient Document Re-Ranking for Transformers by Precomputing Term Representations. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 49–58.

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. *arXiv preprint arXiv:1611.09268*.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *arXiv preprint arXiv:1901.04085*.

Ashwin Paranjape, Omar Khattab, Christopher Potts, Matei Zaharia, and Christopher D Manning. 2021. Hindsight: Posterior-guided Training of Retrievers for Improved Open-Ended Generation. *arXiv preprint arXiv:2110.07752*.

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.

Ruiyang Ren, Shangwen Lv, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021a. PAIR: Leveraging Passage-centric Similarity Relation for Improving Dense Passage Retrieval. *arXiv preprint arXiv:2108.06027*.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021b. RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking. *arXiv preprint arXiv:2110.07367*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models. *arXiv preprint arXiv:2104.08663*.

Benchang Wei, Tao Guan, and Junqing Yu. 2014. Projected Residual Vector Quantization for ANN Search. *IEEE multimedia*, 21(3):41–51.

Ji Xin, Chenyan Xiong, Ashwin Srinivasan, Ankita Sharma, Damien Jose, and Paul N Bennett. 2021. Zero-Shot Dense Retrieval with Momentum Adversarial Domain Invariant Representations. *arXiv preprint arXiv:2110.07581*.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. In *International Conference on Learning Representations*.

Ikuya Yamada, Akari Asai, and Hannaneh Hajishirzi. 2021. Efficient passage retrieval with hashing 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 2: Short Papers)*, pages 979–986, Online. Association for Computational Linguistics.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021a. Jointly Optimizing Query Encoder and Product Quantization to Improve Retrieval Performance. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 2487–2496.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021b. Learning Discrete Representations via Constrained Clustering for Effective and Efficient Dense Retrieval. *arXiv preprint arXiv:2110.05789*.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020a. Learning to retrieve: How to train a dense retrieval model effectively and efficiently. *arXiv preprint arXiv:2010.10469*.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020b. Repbert: Contextualized text embeddings for first-stage retrieval. *arXiv preprint arXiv:2006.15498*.