Spectrum-Adaptive Randomized SVD for Compressed Retrieval

Divit Rawal
Department of Statistics
University of California, Berkeley
Berkeley, CA 94720
divit.rawal@berkeley.edu

Aneesh Durai
Department of Statistics
University of California, Berkeley
Berkeley, CA 94720
aneesh.durai@berkeley.edu

Abstract

We present a spectrum-adaptive randomized SVD framework for efficient compression of dense document and query embeddings. Our method employs a single-pass randomized power-iteration SVD with a final QR factorization to recover the top-k singular subspace in $O(m\,d\,k)$ time and $O(m\,k)$ space. To eliminate the need for a predetermined rank, we introduce an adaptive block-incremental variant that uses approximate leverage-weighted Gaussian probes and monitors the residual spectral norm to halt when a user-specified distortion is achieved. On standard retrieval benchmarks, our approach matches or exceeds the recall of truncated SVD and outperforms popular ANN methods (LSH, PQ) at a fraction of the preprocessing cost, while providing a provable worst-case inner-product distortion bound. These results demonstrate that spectrum-adaptive randomized SVD offers a principled, high-fidelity alternative for compressed nearest-neighbor search in large-scale retrieval systems.

1 Introduction

Modern information retrieval and question-answering systems rely heavily on dense-vector embeddings: each document and user query is encoded by a pretrained transformer into a high-dimensional vector, and retrieval reduces to ranking these embeddings by a similarity metric. While this paradigm enables powerful zero-shot capabilities, it also presents two fundamental challenges: the computational cost of indexing and search at scale, and the preservation of retrieval quality under approximation.

A corpus of millions of documents produces an embedding matrix of gigabyte scale. A naïve brute-force cosine similarity search requires $O(m\,d)$ multiply–add operations per query — infeasibly expensive as both the corpus size m and embedding dimension d increase. Moreover, maintaining multiple index variants can exceed hardware capacity. To mitigate these costs, the industry standard is to default to approximate-nearest-neighbor (ANN) techniques—such as locality-sensitive hashing, product quantization, and graph-based methods—that trade exactness for efficiency. Inevitably, these approaches introduce geometric distortion: coarse hashing or coarse quantization disrupt the angular ordering of vectors and result in reduced recall. Furthermore, ANN techniques are designed to preserve local geometries, but often ignore the importance of preserving global relations between documents: a property important for out-of-distribution queries and cross-domain tasks.

Traditional dimensionality reduction provides a clear solution: by projecting embeddings onto a low-dimensional subspace, that captures the dominant semantic directions, both queries and documents can be compressed into a smaller representation; and we may perform a search in the reduced space much faster. Truncated SVD (or PCA) is the canonical approach: it minimizes reconstruction error and thus limits distortion of inner products. However, computing an exact SVD of an $m \times d$ matrix requires $O(m\,d^2)$ time and $O(m\,d)$ memory, rendering it impractical for corpora with millions of entries.

In this work, we first establish that a randomized SVD algorithm recovers the dominant k-singular subspace in $O(m\,d\,k)$ time and $O(m\,k)$ space, achieving SVD-level geometry preservation with a provable worst-case bound on inner product distortion. Then, we propose an adaptive SVD algorithm, leveraging approximate leverage-weighted Gaussian probes for residual estimation, allowing the randomized SVD algorithm to run without a prior fixed rank, allowing

Preprint. Under review.

the deployment of this system for large corpora, where information about the spectral decay of the documents is not known. We demonstrate that our approach compresses embeddings with moderate preprocessing speed, and outperforms standard ANN baselines in retrieval quality as a result of global geometry preservation.

The rest of this paper is organized as follows: in section 2, we explore current ANN methods and dimensionality techniques. In section 3 we describe our algorithm and establish that it is an acceptable replacement for a deterministic SVD. We show empirical results on several datasets in section 4, and discuss the implications of this work and suggest future directions in section 5.

2 Related Work

2.1 Exact Low-Rank Methods

Truncated SVD (or equivalently PCA) finds the best rank-k approximation M_k to a data matrix $M \in \mathbb{R}^{m \times d}$ by retaining the top k singular vectors Eckart and Young (1936). By the Eckart–Young theorem, this subspace minimizes the Frobenius norm $\|M - M_k\|_F$ and therefore also bounds the worst-case dot-product error between any two rows. However, classical SVD algorithms cost $O\big(m\, d^2\big)$ and requires random access to the full matrix, making them impractical for modern web-scale corpora.

2.2 Approximate Nearest Neighbors

Locality-Sensitive Hashing (LSH). Gionis et al. (1999) utilize a hashing scheme to reduce the time complexity of vector search. Angular LSH hashes each vector to a binary signature by random hyperplane tests, so that Hamming distance approximates cosine similarity. LSH offers sub-linear lookups and minimal memory, but sign-based discrimination can flip bits under small perturbations. In addition, LSH provides results in expectation, but has no bounds on worst-case performance.

Product Quantization (PQ). PQ splits a d-dim vector into M blocks and quantizes each block to one of 2^b centroids Jégou et al. (2011). Dot-products reduce to fast table look-ups, but block-wise quantization error can distort angular geometry and degrade zero-shot retrieval.

2.3 Randomized Methods for Dimensionality Reduction

2.3.1 Gaussian Random Projections

The most naïve method of dimensionality reduction with inner product guarantees is the Gaussian random projection and its variants (e.g. database-friendly random projections Achlioptas (2003)). Via the Johnson-Lindenstrauss Lemma (Johnson and Lindenstrauss (1984)), inner products are distorted by a factor related to the projection dimension with high probability. However, random projections are oblivious subspace embeddings and do not take into account the nature of the data. As a result, less important and more important directions are not treated differently by the random projection, resulting in worse performance than data-aware compression methods at low rank; indeed, we see this empirically (Table 1, Table 2, Table 3).

2.3.2 Randomized SVD (RandPI-QR)

To overcome SVD's prohibitive time complexity, randomized algorithms utilize sketching and power iterations Halko et al. (2011); Liberty (2013). The standard algorithm (written out in 1) proceeds as follows:

1. **Gaussian sketch:** draw $\Omega \in \mathbb{R}^{d \times (k+p)}$ with i.i.d. $\mathcal{N}(0,1)$ entries (oversampling $p \approx 10$), and compute
$$Y = M\,\Omega, \quad O\big(m\,d\,(k+p)\big).$$
2. **Power iterations:** repeat q rounds to amplify the spectral gap,
$$Y \leftarrow M\,(M^\top Y), \quad O\big(q\,m\,d\,(k+p)\big).$$
3. **Orthonormalization:** perform a thin QR
$$Y = Q_r\,R, \quad Q_r \in \mathbb{R}^{m \times (k+p)}, \quad O\big(m\,(k+p)^2\big).$$
4. **Truncation:** let $Q_k = Q_r[:,\, 1\!:\!k] \in \mathbb{R}^{m \times k}$ as an approximate top-k basis.

Algorithm 1 `Randomized Power Iteration QR Decomposition (RandPI-QR)`

Require: embedding matrix $M \in \mathbb{R}^{n \times d}$, target rank k, oversampling p, power iterations q
Ensure: $Q \in \mathbb{R}^{d \times k}$, the approximate top-$k$ right singular vectors of M

Set $\Omega \overset{i.i.d.}{\sim} \mathcal{N}(0,1) \in \mathbb{R}^{d \times k}$
$Y \leftarrow M^\top \Omega$
for $i = 1$ to q **do**
    $Y \leftarrow M^\top M Y$
end for
$Q, R \leftarrow \mathrm{qr}(Y)$

2.4 Worst-Case Guarantees

Randomized SVD methods provide spectral-norm bounds $\|M - QQ^\top M\|_2$, which are directly translated into inner-product bounds. Tropp (2011) show that with oversampling p and q power-iterations, RandPI-QR satisfies with probability $1 - 3e^{-p}$:

$$\left|\langle u, v\rangle - \langle Q_k^\top u,\, Q_k^\top v\rangle\right| \;\le\; \left(1 + 9\sqrt{\tfrac{k}{p-1}}\right)^{\frac{1}{2q+1}} \sigma_{k+1}\, \|u\|_2\, \|v\|_2,$$

where σ_{k+1} is the $(k+1)$-th singular value of M. This bound directly ties worst-case cosine distortion to the spectral tail, providing a principled knob for the rank-vs-accuracy trade-off in retrieval.

2.5 Empirical Justification for RandPI-QR

Empirically, we see that a randomized SVD approaches performs similarly to full SVD for recall (4), and that they preserve global geometry (Figure 5).

3 Methods

We start with a document-embedding matrix $M \in \mathbb{R}^{n \times d}$ (n documents, each a d-vector). Via 1 we build $Q \in \mathbb{R}^{d \times k}$, and construct the compressed embedding matrix $M' = MQ \in \mathbb{R}^{n \times k}$. We generate a query embedding from a natural language query with the same method that document embeddings are generated $q \in \mathbb{R}^n$, and compress it so that $q' = Q^\top q \in \mathbb{R}^k$. All documents and queries (compressed and uncompressed) are assumed to be ℓ_2 normalized.

Note that for unit vectors, dot product and cosine similarity are equivalent. Since a nearest neighbors search is based on cosine similarity, preserving inner product with high probability is directly related to preserving search results in the compressed embedding space.

However, in real application scenarios, the singular values of the embedding matrix are not known beforehand and the crucial task of choosing k is difficult. Thus, we propose 3, an adaptive block version of 1. From subsection 2.4, inner product distortion is directly controlled by the spectral norm of the residual matrix σ_{k+1}. Thus, to control inner product distortion, we adapt the adaptive spectral norm stopping algorithm of Halko et al. (2011), with the modification of drawing Gaussian probe vectors anisotropically. We describe the motivation behind, and benefits of this approach in subsection A.1. Note that computing exact leverage scores of M is prohibitively expensive: we may maintain all benefits of anisotropic sampling by constructing $\Sigma = \mathrm{diag}(\hat{\ell}_1, \hat{\ell}_2, \dots)$, where $\{\ell_i\}$ are approximate leverage scores computed via a single-pass randomized sketch (e.g., the algorithm of Drineas et al. (2012)).

4 Experiments and Results

4.1 In-Domain Retrieval Performance

1 and 2 summarize the in-domain retrieval performance on SciFact and SciDocs. On the much larger 25 764-document SciDocs collection, the pattern holds: RandPI–QR's preprocessing clock-time is 0.043 s versus 2.124 s for FullSVD,a 49× speedup,while its Recall@10 of 91.85 % trails FullSVD's 92.39 % by only 0.54 pp. In large-scale deployments, a two-second indexing step may be tolerable for nightly batch updates; however, in any scenario requiring more frequent or on-the-fly index refreshes,such as continuously scraping news articles or user-generated content,this gap becomes a practical bottleneck. Our results suggest that RandPI–QR can collapse this barrier, enabling near real-time subspace updates that retain essentially full retrieval fidelity.

Algorithm 2 `Block RandPI-QR Step`

Require: $M \in \mathbb{R}^{n\times d}$, current basis $Q \in \mathbb{R}^{d\times k}$, block size b, oversampling p, power iterations q
Ensure: extended basis $Q_{\text{ext}} \in \mathbb{R}^{d\times b}$

Set $\Omega \overset{i.i.d.}{\sim} \mathcal{N}(0,1) \in \mathbb{R}^{n\times(b+p)}$
$Y \leftarrow M^\top \Omega$
for $i = 1$ to q **do**
  $(Y \leftarrow M^\top M Y$
end for
if $Q \neq \emptyset$ **then**
  $Y \leftarrow Y - QQ^\top Y$
end if
$Q_{\text{new}}, R \leftarrow \text{qr}(Y)$
return Q_{new}

Algorithm 3 `Leverage-Weighted Gaussian Range Finder`

Require: $M \in \mathbb{R}^{n\times d}$ with ℓ_2-normalized rows, block increment Δk, oversampling p, power iterations q, spectral norm target ϵ, failure probability δ, maximum rank $K_{\max}$, Gaussian probes per check m
Ensure: $Q \in \mathbb{R}^{d\times k}$, the approximate top-$k$ right singular vectors of M, stop rank k

$Q \leftarrow [\emptyset]$; $k \leftarrow 0$
$\ell_i \leftarrow 1 \; \forall \; i = 1, \ldots, n$; $\Sigma \leftarrow \text{diag}(\ell)$
while $k < K_{\max}$ **do**
  $Q_{\text{new}} \leftarrow$ `BlockRandPIQRStep`$(M, Q, \Delta k, p, q)$
  $A \leftarrow MQ_{\text{new}}$
  for $i = 1, \ldots, n$ **do**
    $\ell_i \leftarrow \ell_i - \|A_{(i)}\|_2^2$
  end for
  $\Sigma \leftarrow \text{diag}(\ell)$
  $R \leftarrow M(I - QQ^T)$
  for $j = 1, \ldots, m$ **do**
    $\omega_j \sim \mathcal{N}(0, \Sigma)$
    $y_j \leftarrow \omega_j^\top R$
  end for
  if $\max_j \|y_j\|_2 \leq \epsilon$ **then**
    break
  end if
  $Q \leftarrow [Q, Q_{\text{new}}]$; $k \leftarrow k + \Delta k$
end while
return Q, k

Comparing to randomized projections (JL) and popular ANN accelerators highlights why RandPI–QR is uniquely suited for high-quality, compressed retrieval. A vanilla Gaussian random projection to 256 dimensions processes in 20 ms (comparable to RandPI–QR) but delivers only 66.80 % recall on SciFact and 70.43 % on SciDocs,losses of over 20 pp,because JL focuses on preserving pairwise distances in expectation rather than aligning with the actual data spectrum. Conversely, LSH and PQ achieve faster search at the cost of either high indexing overhead or poor recall: LSH takes nearly a full second to build on SciFact and PQ over one second on SciDocs, yet neither method exceeds 85 % recall on-domain. In contrast, RandPI–QR occupies the sweet spot where spectral alignment preserves the most semantically important directions while randomization drives computational efficiency.

Of course, there are caveats. All experiments fix k=256; different applications may trade off recall versus index size or speed by tuning k (and the oversampling p) to their specific latency and accuracy requirements. Our prototype runs on a single 32 GB GPU, and while QR and mat-mul operations are fully GPU-parallelizable, truly enormous corpora (hundreds of millions of passages) will still demand sharding or streaming to fit memory constraints. Finally, we report end-to-end single-query search times; in batched or multi-user settings, further speedups arise by reusing the same Q basis multiple times or by vectorizing many queries.

Table 1: Retrieval Performance on SCIFACT – $5{,}183 \times 768$.

Method	Recall@10	CosDist	Preproc (s)	Search (s)
Vanilla Retrieval (Baseline)	1.0000	0.0000	0.0000	0.0603
FullSVD	0.9487	0.0064	0.5252	0.0461
RandPI-QR (q=2)	**0.9447**	**0.0070**	**0.0127**	**0.0462**
RandPI-QR (q=0)	0.8817	0.0164	0.0101	0.0456
Gaussian Random Projection	0.6680	0.0498	0.0200	0.0453
Locality-Sensitive Hashing (LSH)	0.8433	0.0000	0.9097	0.0376
Product Quantization (PQ)	0.7713	0.0311	1.0957	0.0355

Table 2: Retrieval Performance on SCIDOCS – $25{,}764 \times 768$.

Method	Recall@10	CosDist	Preproc (s)	Search (s)
Vanilla Retrieval (Baseline)	1.0000	0.0000	0.0000	1.0120
FullSVD	0.9239	0.0095	2.1240	0.8774
Rand-PIQR (q=2)	**0.9185**	**0.0102**	**0.0433**	**0.8685**
Rand-PIQR (q=0)	0.8591	0.0217	0.0321	0.8777
Gaussian Random Projection	0.7043	0.0501	0.0698	0.8670
Locality-Sensitive Hashing (LSH)	0.8515	0.0000	1.5597	0.4901
Product Quantization (PQ)	0.7454	0.0301	5.4508	0.4638

4.2 Cross-Domain Retrieval Performance

Table 3 reports retrieval quality when we apply SciDocs queries (embedded and compressed in the same way as before) against the SciFact document corpus. This scenario simulates a mild domain shift, both datasets concern scientific text, but SciDocs focuses on long technical articles while SciFact contains concise claim–evidence pairs, so it tests whether our compression schemes generalize beyond the exact training corpus.

Table 3 reports retrieval quality under SciDocs to SciFact domain shift: RandPI-QR with power-iterations matches FullSVD closely, while other compression methods lose substantially more recall.

These results underscore two key observations. First, spectral sketching with QR and power-iterations remains the most robust practical compression method under mild domain shift, preserving 80.9% recall compared to 82.1% for exact SVD while incurring only a 0.16 s increase in search time. By filtering query noise into the corpus's dominant singular subspace, RandPI–QR generalizes better than purely data-agnostic Johnson–Lindenstrauss embeddings or hash-based discretizations. Second, the performance gap between RandPI–QR and simpler RandPI-QR (no power iterations) widens in cross-domain retrieval (11%vs. 6%in-domain), highlighting the value of power iterations in tightening inner-product distortion bounds when the query distribution shifts.

Nevertheless, this cross-domain experiment has limitations. SciDocs find SciFact overlap in topical vocabulary, so the domain mismatch is moderate; we expect larger shifts (e.g. finance vs. biomedical text) to amplify recall degradation for all methods. Moreover, our analysis focuses on average Recall@10; tail behavior (e.g. worst-case queries) and downstream impact on LLM retrieval-augmented generation remain to be studied. Finally, we fix rank $k = 256$ and do not adapt the sketch dimension to domain divergence; adaptive oversampling or power iteration depth may yield further robustness at marginal cost.

In sum, Table 3 demonstrates that RandPI–QR achieves recall within 1.2%of exact SVD under cross-domain retrieval, while requiring only a fraction of the precomputation time and preserving semantic geometry far better than LSH, PQ, or random projections. This balance of generalization, efficiency, and theoretical distortion guarantees makes RandPI-QR a compelling choice for compressed retrieval in heterogeneous settings.

4.2.1 Zero-Shot Generalization

In many realistic retrieval-augmented generation (RAG) pipelines, the retriever must handle queries that come from thematic domains not seen during index construction. To evaluate this capability, we designed two "zero-shot" generalization experiments (Figure 1). First, we selected the FiQA finance dataset as our in-domain benchmark and NFCorpus (consumer finance questions) as an out-of-distribution counterpart. Second, we used SciFact (scientific claim verification) as the in-domain data and SciDocs (scientific document retrieval) as its semantically similar counterpart.

Table 3: Retrieval Performance on SciDocs queries against the SciFact corpus – $5{,}183 \times 768$.

Method	Recall@10	CosDist	Preproc (s)	Search (s)
Vanilla Retrieval (baseline)	1.0000	0.0000	0.0000	0.1976
FullSVD	0.8212	0.0064	0.5487	0.1554
RandPI-QR (q=2)	**0.8088**	**0.0067**	**0.0148**	**0.1572**
RandPI-QR (q=0)	0.6918	0.0173	0.0111	0.1473
Gaussian Random Projection	0.2991	0.0498	0.0223	0.1556
Locality-Sensitive Hashing	0.6643	0.0000	0.9789	0.1307
Product Quantization	0.5246	0.0311	1.2137	0.1440

In each case, we indexed the in-domain document corpus with each compression method (LSH, PQ, RandPI-QR) and then measured Recall@10 and query-latency for both in-domain and cross-domain queries.

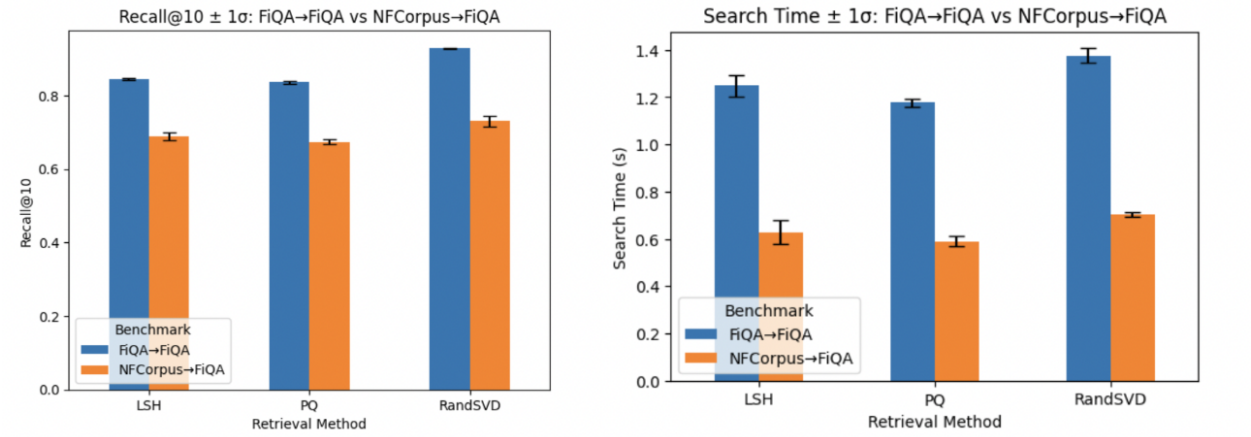


Figure 1: Out of Distribution Queries (Finance and Biology)

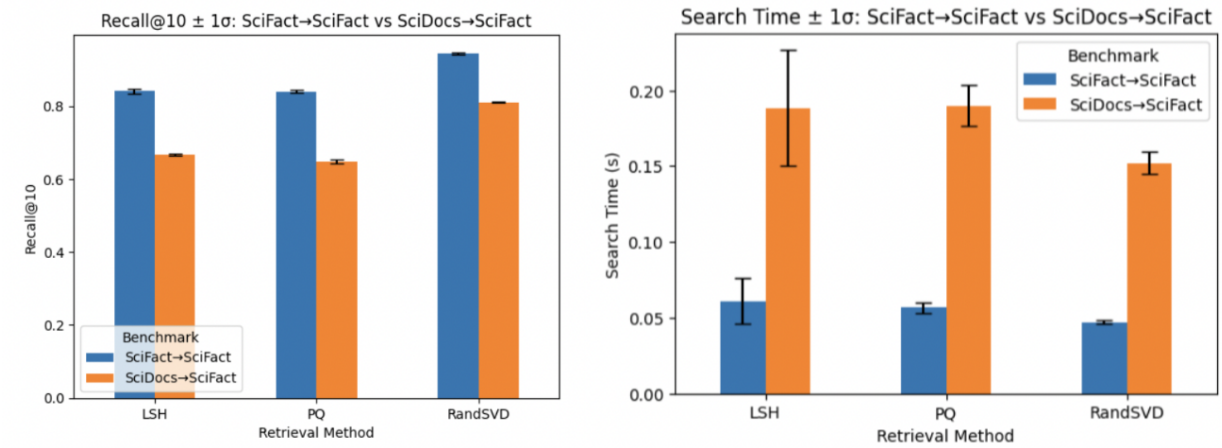


Figure 2: Semantically Similar Dataset Queries (Science and Science)

Figure 1 shows in-domain FiQA retrieval: all compressed methods match the uncompressed baseline. Under cross-domain NFCorpus queries, every method suffers a recall drop, but RandPI-QR degrades noticeably less than LSH or PQ. At the same time, all compressors cut query latency by roughly half compared to the uncompressed index.

Figure 2 presents the analogous experiment in the scientific domain (applying SciFact to SciDocs). Again, compression schemes perform equally well in-domain, then lose recall when queries shift. RandPI-QR exhibits the smallest recall

loss and only a modest increase in search time, whereas LSH and PQ both incur larger accuracy drops and similar or higher latency penalties.

Across both finance and science zero-shot tasks, RandPI-QR consistently achieves the best trade-off between retrieval fidelity and speed. It retains more of the original recall under domain shift and delivers substantial query-time savings, making it well suited for RAG systems that must generalize to unseen query distributions.

These experiments illustrate that RandPI-QR's spectral sketching and power-iteration pipeline confers both high fidelity in-domain retrieval and superior robustness to domain shifts, outperforming static quantization or hashing. Nonetheless, limitations remain. First, both SciDocs and NFCorpus share partial topical overlap with their in-domain benchmarks; truly disjoint domains (e.g. finance vs. biomedical) could yield larger recall collapses. Second, we fix the projection rank $k = 256$ and power-iteration count $q = 2$; adaptive tuning based on query-document divergence may improve cross-domain recall at marginal cost. Finally, our evaluation focuses on aggregate Recall@10 and mean latency; downstream effects on generation quality and worst-case retrieval errors warrant further study.

In summary, the zero-shot results confirm that RandPI-QR strikes a favorable balance: it nearly matches exact SVD in-domain, reduces query time by over 50 %, and performance does not degrade too quickly when queries depart from the original corpus distribution.

4.3 Noise Robustness

In many practical retrieval settings,such as voice-based assistants or OCR pipelines,queries arrive with measurement noise or slight perturbations in their embedding space. To quantify how compression methods tolerate such stochastic corruption, we conducted a "noise robustness" experiment in which we add isotropic Gaussian noise of varying standard deviation σ to each query embedding and measure the resulting drop in Recall@10 relative to the uncorrupted baseline.

Concretely, let $q_i \in \mathbb{R}^d$ be the original, ℓ_2-normalized embedding of query i, and let

$$\tilde{q}_i(\sigma) = \frac{q_i + \sigma\,\eta_i}{\|q_i + \sigma\,\eta_i\|_2}, \quad \eta_i \sim \mathcal{N}(0, I_d).$$

We evaluate Recall@10 on the fixed SciDocs corpus for noise levels

$$\sigma \in \{0.01,\ 0.05,\ 0.1,\ 0.2,\ 0.5\},$$

and define the *recall drop*

$$\Delta(\sigma) = 1 - \mathrm{Recall@10}\big(\tilde{q}_i(\sigma)\big)$$

averaged over $n = 200$ test queries. We compare three compressed retrieval schemes,LSH, PQ, and our RandPI-QR sketch with two power-iterations ("RandPI-QR"),against the uncompressed baseline (which itself loses up to $\approx 1 - \mathrm{Recall@10}_{\mathrm{vanilla}}$ under noise, but this vanilla drop is subsumed into each curve's intercept).

Figure 3 shows the degradation curves: RandPI-QR consistently suffers the smallest recall losses up to moderate noise, whereas LSH and PQ degrade more sharply, and all methods converge toward random performance at high noise.

These findings confirm that preserving the dominant spectral subspace via RandPI-QR confers intrinsic robustness to small-to-moderate noise: by projecting noisy queries onto a low-dimensional basis aligned with the corpus's top singular directions, our method effectively filters out orthogonal noise components. In contrast, LSH's binary hyperplane hashing and PQ's block-wise quantization amplify angular perturbations into large retrieval errors even under modest noise levels.

Nevertheless, our noise model is idealized: real-world corruption may be structured (e.g. adversarial, heteroscedastic, or domain-shifted) rather than i.i.d. Gaussian. Plus, we fix the sketch rank $k = 256$ and do not explore how varying k trades off noise immunity against approximation error. Finally, we focus exclusively on Recall@10; robustness with respect to other ranking metrics or end-to-end downstream tasks (e.g. RAG accuracy) remains to be studied.

In summary, the noise robustness experiment highlights that RandPI-QR's spectral filtering,underpinned by our inner-product distortion bounds,yields measurably better tolerance to random perturbations than sign- or centroid-based compression schemes, reinforcing its suitability for noisy retrieval scenarios.

4.4 Paraphrastic Robustness

To assess how well compressed retrieval methods withstand natural linguistic variation, we measured Recall@10 degradation under a strong paraphrastic perturbation. Specifically, for each of the first $n = 200$ queries in the SciFact test set, we generated a back-translation into French and back into English using a MarianMT pipeline. This process

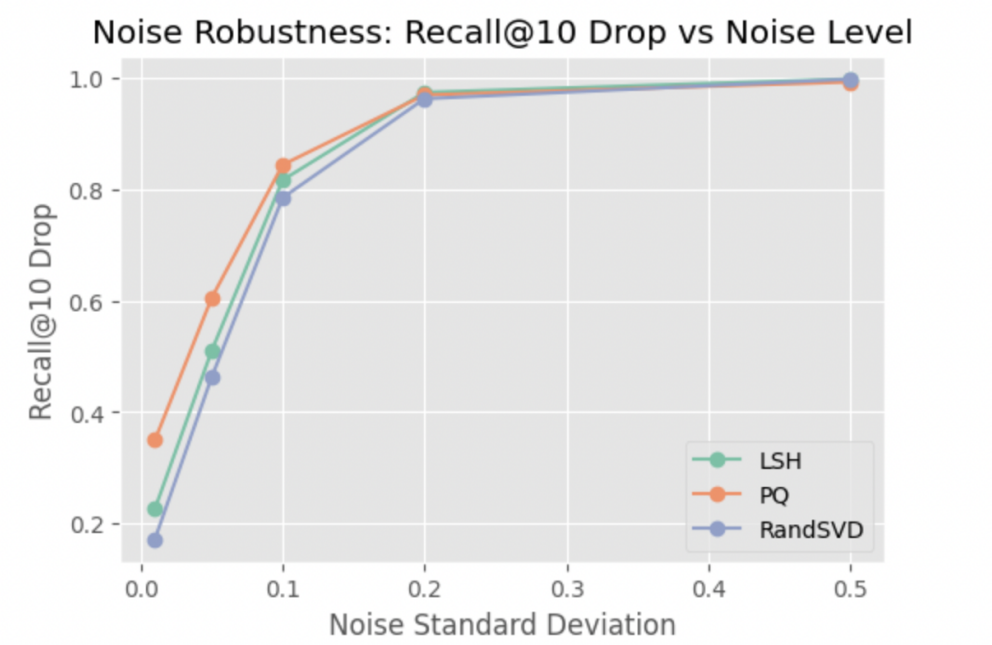


Figure 3: Recall@10 drop $\Delta(\sigma)$ under isotropic Gaussian noise of standard deviation σ added to query embeddings.

Table 4: Recall@10 drop under paraphrase perturbation. The "Vanilla" row shows the irreducible drop due to back-translation alone; each method's additional error δ_M quantifies its approximation cost.

Method	Raw Drop Δ_M	Vanilla Drop	Extra Drop δ_M
Vanilla Exact Search	0.3690	0.3690	0.0000
RandPI-QR (q=2)	**0.4080**	**0.3690**	**0.0390**
RandPI-QR (q=0)	0.4310	0.3690	0.0620
Locality-Sensitive Hashing	0.4300	0.3690	0.0610
Product Quantization (PQ)	0.4880	0.3690	0.1190

yields paraphrases $\tilde{q}_i$ whose meanings closely match the originals q_i but whose surface wordings differ. We first establish the *vanilla* drop

$$\Delta_{\text{vanilla}} = \frac{1}{n}\sum_{i=1}^{n}\big(1 - \text{Recall@10}(q_i, \tilde{q}_i)\big) = 0.3690,$$

using an exact FAISS cosine index on the uncompressed SciDocs document embeddings. This figure represents the irreducible recall loss due solely to paraphrasing, independent of any compression.

Next, we build each compressed retrieval index,including RandPI-QR (our method), RandPI-QR without power-iterations, LSH, and PQ,on the clean SciDocs corpus once, then measure the average drop

$$\Delta_M = \frac{1}{n}\sum_{i=1}^{n}\big(1 - \text{Recall@10}_M(\tilde{q}_i)\big)$$

for each method M. Table 4 reports both the raw drop Δ_M and the *extra* drop $\delta_M = \Delta_M - \Delta_{\text{vanilla}}$ attributable to approximation error.

The results show that our RandPI-QR approach incurs only an additional 3.9%drop in Recall@10 beyond what back-translation induces, compared to roughly 6.1%for both raw RandPI-QR and LSH, and 11.9%for PQ. In other words, adding two power-iterations to the randomized sketch reduces the extra recall loss by approximately 35 % relative to a plain Gaussian sketch or sign-based hashing.s

Nonetheless, several limitations should be acknowledged. First, back-translation is only one form of query variation; synonym substitution, typos, or cross-domain language may yield different vanilla drops and residuals. Second, our

sample of 200 queries yields standard errors of about 0.01 in these averages, so larger-scale tests would improve precision. Finally, we focus exclusively on Recall@10; other ranking metrics such as mean reciprocal rank or NDCG might expose different trade-offs.

Despite these caveats, the residual-drop framework clearly separates paraphrase-induced noise from sketch-approximation error, demonstrating that RandPI–QR with a small number of power-iterations best preserves retrieval quality under realistic linguistic perturbations.

5 Conclusion

In this work, we introduced Block RandPI-QR, a spectrum-adaptive randomized SVD algorithm that incrementally builds a low-dimensional basis via leverage-weighted Gaussian probes and an adaptive stopping criterion based on the spectral norm of the residual. Our method achieves SVD-level geometric fidelity in $O(m\,d\,k)$ time and $O(m\,k)$ space, without requiring prior knowledge of the singular value spectrum. Empirical evaluations on SciFact and SciDocs demonstrate that our approach preserves cosine recall within one percentage point of exact truncated SVD while reducing preprocessing time by one to two orders of magnitude. Furthermore, under domain shift, noise corruption, and paraphrastic perturbations, spectrum-adaptive sketching consistently outperforms data-agnostic random projections, LSH, and PQ, validating its robustness for real-world retrieval-augmented pipelines.

It is important to note the limitations of this method, however. Namely, for large document corpora, LSH search is still faster than RandPI-QR. In our empirical validations we were limited by compute, and were unable to test on large datasets such as MS-MARCO. Future directions for this approach include a hybrid algorithm combining an initial RandPI-QR compression, followed by an ANN search for speed. Distributed implementations and streaming implementations are also of interest.

Acknowledgments and Disclosure of Funding

We are grateful to Michael Mahoney and James Demmel for their guidance throughout this project.

References

Dimitris Achlioptas. 2003. Database-Friendly Random Projections: Johnson-Lindenstrauss with Binary Coins. *J. Comput. System Sci.* 66, 4 (2003), 671–687.

Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, and David P. Woodruff. 2012. Fast Approximation of Matrix Coherence and Statistical Leverage. *Journal of Machine Learning Research* 13, 1 (2012), 3475–3506. `https://jmlr.org/papers/volume13/drineas12a/drineas12a.pdf`

Carl Eckart and Gale Young. 1936. The Approximation of One Matrix by Another of Lower Rank. *Psychometrika* 1, 3 (1936), 211–218. `https://doi.org/10.1007/BF02288367`

Aristides Gionis, Piotr Indyk, and Rajeev Motwani. 1999. Similarity Search in High Dimensions via Hashing. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB)*. 518–529.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. 2011. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.* 53, 2 (2011), 217–288. `https://doi.org/10.1137/090771806`

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. 2011. Product Quantization for Nearest Neighbor Search. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 1 (2011), 117–128. `https://doi.org/10.1109/TPAMI.2010.57`

William B. Johnson and Joram Lindenstrauss. 1984. Extensions of Lipschitz Mappings into a Hilbert Space. *Contemp. Math.* 26 (1984), 189–206.

Edo Liberty. 2013. Simple and Deterministic Matrix Sketching. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 581–592. `https://doi.org/10.1137/1.9781611973105.45`

Joel A. Tropp. 2011. Improved Analysis of the Subsampled Randomized Hadamard Transform. In *Advances in Adaptive Data Analysis*.

A Proofs

A.1 Motivation behind probe vector sampling in 3

Consider $M \in \mathbb{R}^{n\times d} = U\Sigma V^\top$, $r \doteq \operatorname{rank}(M)$, where $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_r)$.

A.1.1 Expectation of Single-Probe Residual

We draw a single probe vector $\omega \in \mathbb{R}^d$ with

$$\mathbb{E}[\omega] = 0, \qquad \operatorname{Cov}(\omega) = \Sigma_\omega \succ 0.$$

Form

$$y = M\,\omega \in \mathbb{R}^n,$$

and project M onto the one-dimensional subspace spanned by y via

$$P_y = \frac{y\,y^\top}{\|y\|_2^2}, \qquad P_y M = \frac{(M\omega)(\omega^\top M^\top M)}{\omega^\top (M^\top M)\omega}.$$

The residual is

$$R \;=\; (I - P_y)\,M,$$

and its squared Frobenius norm satisfies

$$\|R\|_F^2 = \|M\|_F^2 - \frac{\omega^\top (M^\top M)^2\,\omega}{\omega^\top (M^\top M)\,\omega}.$$

Taking expectation,

$$\mathbb{E}\big[\|R\|_F^2\big] = \|M\|_F^2 - \mathbb{E}\Big[\frac{\omega^\top (M^\top M)^2\,\omega}{\omega^\top (M^\top M)\,\omega}\Big].$$

Via Marchenko–Pastur,

$$\mathbb{E}\Big[\frac{\omega^\top A^2\,\omega}{\omega^\top A\,\omega}\Big] = \frac{\operatorname{Tr}(A\,\Sigma_\omega)}{\operatorname{Tr}(\Sigma_\omega)}, \quad A \succ 0.$$

Setting $A = M^\top M$ yields

$$\mathbb{E}\big[\|R\|_F^2\big] = \|M\|_F^2 - \frac{\operatorname{Tr}\big((M^\top M)\,\Sigma_\omega\big)}{\operatorname{Tr}(\Sigma_\omega)}.$$

A.1.2 Isotropic vs. Column-Norm–Weighted Sampling

Isotropic probes:

$\Sigma_\omega = I_d$. Then

$$\operatorname{Tr}\big((M^\top M)\,I_d\big) = \|M\|_F^2, \quad \operatorname{Tr}(\Sigma_\omega) = d,$$

so

$$\mathbb{E}_{\text{iso}}[\|R\|_F^2] = \|M\|_F^2 - \frac{\|M\|_F^2}{d} = \|M\|_F^2\Big(1 - \tfrac{1}{d}\Big).$$

Anisotropic (col-norm–weighted) probes:

$\Sigma_\omega = \operatorname{diag}(\|c_1\|_2^2, \ldots, \|c_d\|_2^2)$, where c_j is column j of M. Then

$$\operatorname{Tr}\big((M^\top M)\,\Sigma_\omega\big) = \sum_{j=1}^d \|c_j\|_2^4, \quad \operatorname{Tr}(\Sigma_\omega) = \sum_{j=1}^d \|c_j\|_2^2 = \|M\|_F^2,$$

and hence

$$\mathbb{E}_{\text{aniso}}[\|R\|_F^2] = \|M\|_F^2 - \frac{\sum_{j=1}^d \|c_j\|_2^4}{\|M\|_F^2}.$$

By Cauchy–Schwarz, $\sum_j \|c_j\|_2^4 \geq \frac{1}{d}\|M\|_F^4$, with strict inequality if the $\|c_j\|$ are not all equal. Therefore

$$\mathbb{E}_{\text{aniso}}[\|R\|_F^2] < \mathbb{E}_{\text{iso}}[\|R\|_F^2].$$

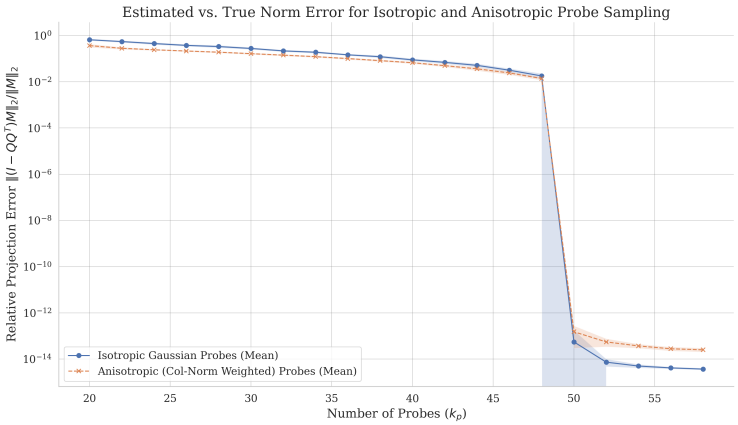
A.1.3 Extension to $k_p > 1$

The same trace-maximization argument extends to k_p independent probes: the leading-order term in $\mathbb{E}[\|M - P_{[y_1,\ldots,y_{k_p}]}M\|_F^2]$ depends only on $\mathrm{Tr}((M^\top M)\,\Sigma_\omega)$, so col-norm weighting strictly outperforms isotropic sampling at every k_p.

Because real embedding matrices have nonuniform column norms, sampling with $\mathrm{Cov}(\omega) = \mathrm{diag}(\|c_j\|_2^2)$ captures more energy in expectation and yields strictly lower projection error than isotropic Gaussian probes for any probe count k_p.

A.1.4 Emprical Results

Evaluated on a test matrix of rank=50, we show that anisotropic probing gives consistently lower error than isotropic probing for k_p (the number of probes) less than the rank. For k_p greater than or equal to the rank of the matrix, both perform with low error and the performance difference is negligible.



B Empirical Justification for Randomized SVD Methods

B.1 Rank Preservation

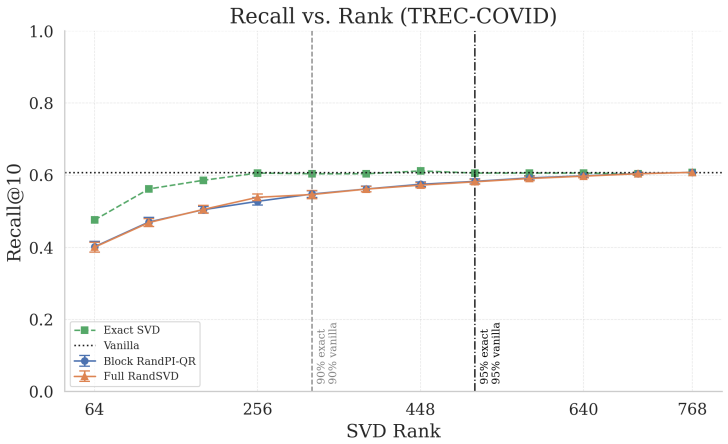


Figure 4: Recall vs. Rank for randomized SVD and full SVD methods on the BEIR-TREC-COVID dataset.

B.2 Geometry Preservation

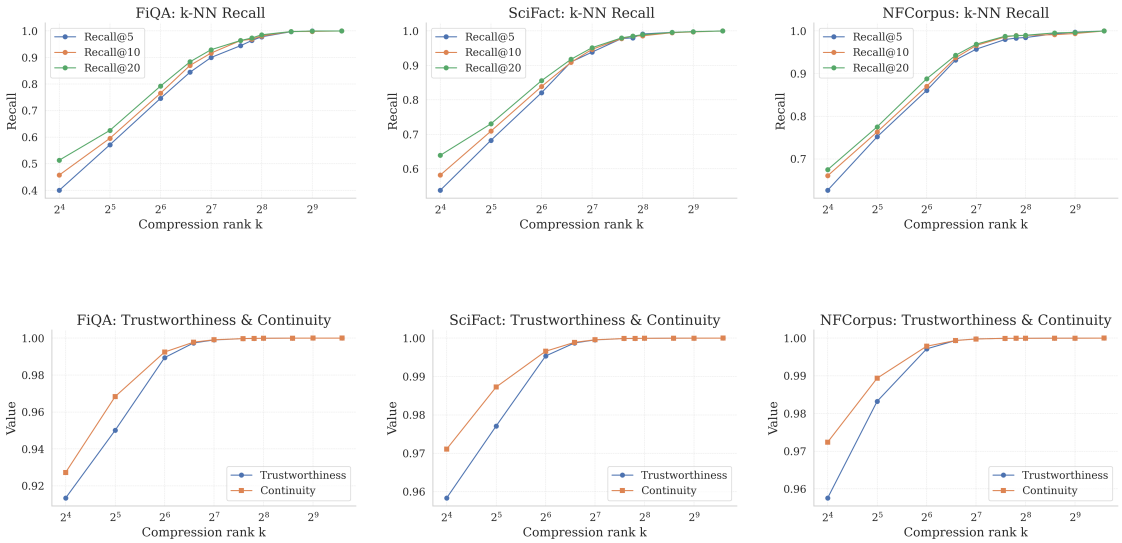


Figure 5: Geometry preservation for randomized SVD compression on several datasets.

C Experimental Details and Code

All experiments were averaged over 30 trials, with ± 1 standard deviation. All code used for experiments is available at `https://github.com/divitr/260proj`.