

Listwise Explanation of Embedding-Based Rankings via Semantic Chunk Grouping

Hyunkyu Kim, Yeeun Yoo, Youngjun Kwak[†]

Financial Tech Lab, KakaoBank Corp.

{conor.k, anny.ye, vivaan.yjkwak}@lab.kakaobank.com

Abstract

Dense embedding rankers score documents through contextual sentence- and passage-level representations, yet listwise explanation methods often attribute rankings to isolated words. We study this mismatch and introduce ChunkGroupSHAP, a listwise Shapley method that clusters semantically related chunks across documents into shared features, preserving contextual evidence while bounding the KernelSHAP regression dimension by the group count. Across MS MARCO, FinanceBench, AILACaseDocs, and FinQA with E5-family rankers and BM25, raw chunks improve rank-reconstruction Fidelity over RankSHAP’s word features in all 11 dense-ranker settings. The best chunk-group configuration further improves on raw chunks in eight of these settings, with the incremental benefit depending on grouping scope; word features remain strongest in three of four BM25 settings. These results show that explanation units should match the ranking model: contextual chunks better suit dense bi-encoders, whereas words remain effective for BM25. ChunkGroupSHAP supports listwise attribution over contextual evidence through a bounded feature space shared across documents.

1 Introduction

Dense embedding retrieval has become the dominant retrieval paradigm for modern semantic search (Thakur et al., 2021; Muennighoff et al., 2023), supporting applications from web retrieval (Karpukhin et al., 2020) to retrieval-augmented generation (Lewis et al., 2020; Guu et al., 2020). As retrieval systems move from lexical matching to dense semantic search, a central interpretability question becomes increasingly important: *why does an embedding model rank documents in a particular order?* In real-world search and RAG systems, users and developers need to

[†]Corresponding author.

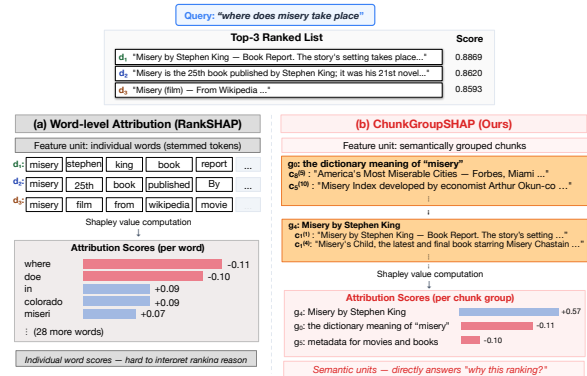


Figure 1: Word-level (RankSHAP) vs. ChunkGroupSHAP (ours) explanation for embedding-based ranking. The descriptive text following each group ID is author-assigned for readability and is not produced by the explainer.

inspect, debug, and audit ranking decisions. Yet ranking explanations have not fully followed the representational shift in retrieval: dense rankers score documents through contextual sentence- or passage-level representations, while many listwise explanation methods still perturb isolated lexical features.

Research on explaining ranking models has progressed from point-wise to listwise approaches. Point-wise methods explain why an individual document is relevant to a query (Ribeiro et al., 2016; Lundberg and Lee, 2017; Choi et al., 2020; Singh and Anand, 2019). In contrast, listwise methods aim to explain why a particular ordering of documents is produced (Chowdhury et al., 2023, 2025; Heuss et al., 2025; Saha et al., 2025). Recent Shapley-value-based listwise methods inherit a lexical view of features: RankSHAP (Chowdhury et al., 2025) uses stemmed words as the explanation unit, and RankingSHAP (Heuss et al., 2025) defines attributions over predefined query-document features. This is a natural choice for lexical rankers such as BM25, whose scores are computed from term

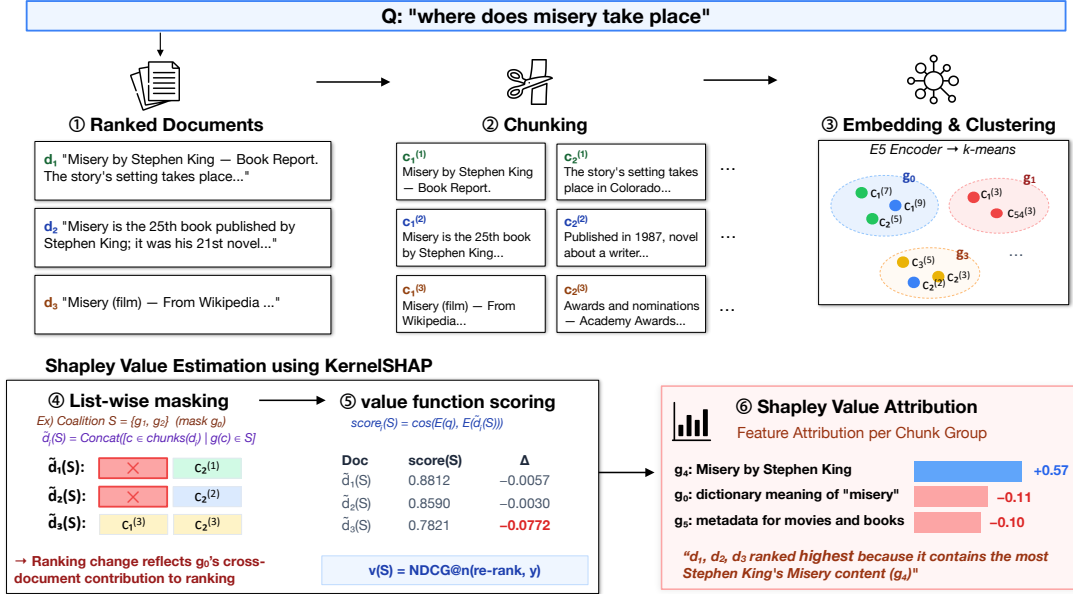


Figure 2: Overview of the ChunkGroupSHAP pipeline for the query “where does misery take place.” Documents are chunked, embedded via E5-small, and clustered into semantic chunk groups that serve as shared cross-document features for listwise Shapley value computation. The descriptive text following each group ID is author-assigned for readability and is not produced by the explainer.

matches. It is a poor fit, however, for dense embedding rankers whose scores are produced from sentence- and passage-level semantic representations (Reimers and Gurevych, 2019; Gao et al., 2021; Li et al., 2023; Wang et al., 2024). The mismatch is both technical and practical: the model being explained operates over contextual semantic evidence, while the explanation shown to a user is often a fragmented list of stemmed token scores. Figure 1(a) illustrates this problem.

A natural response is to explain dense rankings with larger units such as sentences or passages, which are closer to the evidence dense rankers encode. But raw chunks break the listwise mechanism: because attribution applies the *same* mask across documents, it relies on features that recur across them—a word like *misery* perturbs every document containing it, whereas a document-specific chunk perturbs only one, collapsing listwise perturbation into point-wise. Raw chunk spaces also grow with document number and length, making them hard to use when length varies.

To address this limitation, we introduce **ChunkGroupSHAP**, a listwise Shapley explanation method designed for dense semantic retrieval. Rather than treating each chunk as an isolated feature, ChunkGroupSHAP embeds chunks and

clusters semantically similar chunks into shared groups. Each group functions as a cross-document semantic feature: masking a group perturbs all documents containing related evidence, preserving the listwise nature of the explanation while operating at a sentence- or passage-level granularity. We study both corpus-level groups, which define reusable features across queries, and query-local groups, which adapt the feature space to the current ranked list. Figure 2 presents the full pipeline, and Figure 1(b) shows how the resulting explanation provides more inspectable evidence than a list of isolated stemmed words.

Our experiments support a setting-specific view of ranking explanation. Across MS MARCO (Bajaj et al., 2018), FinanceBench (Islam et al., 2023), AILACaseDocs (Bhattacharya et al., 2020), and FinQA (Chen et al., 2021), with BM25 (Robertson and Zaragoza, 2009) and the E5 model family (Wang et al., 2024), we separate the effect of contextual units from that of grouping. Replacing RankSHAP’s word features with raw chunks improves rank-reconstruction Fidelity in all 11 tested E5 settings and produces the largest step in each comparison. The best chunk-group configuration further exceeds raw chunks in eight settings, but matched semantic-versus-random diagnostics show that this additional grouping effect

depends on scope: semantic grouping is higher in the tested query-local settings with nontrivial merging, whereas corpus-level results are mixed. Word features remain strongest in three of four BM25 settings. The explanation unit should therefore follow both the ranker and the retrieval corpus rather than be fixed in advance.

Our contributions are as follows:

- We formulate explanation-unit design as a first-class problem in listwise ranking explanation and compare word, raw-chunk, and chunk-group features for dense and lexical rankers.
- We propose ChunkGroupSHAP, a listwise Shapley method that clusters embedded chunks into shared cross-document features, bounds the regression dimension by the number of groups, and supports corpus-level and query-local variants.
- Across four benchmarks, we show that contextual chunks drive the reconstruction gains for the tested E5-family rankers, word features remain strong for BM25, and the additional Fidelity effect of semantic grouping is setting-dependent.

2 Related Works

Explainable Information Retrieval. Early work on explaining search results followed a pointwise view—explaining why a single document is relevant to a query—using model-agnostic and attribution-based methods such as LIME (Ribeiro et al., 2016), SHAP (Lundberg and Lee, 2017), DeepSHAP (Fernando et al., 2019), and gradient-based attribution (Choi et al., 2020). Since search systems return ordered lists rather than isolated scores, later work targeted listwise explanation (Yu et al., 2022; Chowdhury et al., 2023). Singh and Anand (2020) modeled ranking intent from perturbed-document preference pairs, while Lyu and Anand (2023) represented listwise explanations with selected simple rankers and expanded-query terms. Yu et al. (2022) generated query-aspect explanations for documents in a ranked list, and Rank-LIME (Chowdhury et al., 2023) extended LIME through correlation-based perturbation and a listwise linear surrogate. These methods establish the need to explain rankings as a whole, but leave open what should count as a perturbable feature in a dense semantic ranker.

Shapley Value-based Ranking Explanation.

The Shapley value (Shapley, 1953) underlies feature attribution in machine learning via SHAP (Lundberg and Lee, 2017). RankingSHAP (Heuss et al., 2025) extends Shapley attribution to listwise ranking using predefined query–document features, while RankSHAP (Chowdhury et al., 2025) defines ranking-specific axioms over stemmed words. Rank-reconstruction metrics recur in listwise ranking explanation. RankLIME (Chowdhury et al., 2023) defines Model Fidelity using Kendall’s τ between the original ranking and the ranking reconstructed from feature attributions, while RankSHAP reports both Fidelity and weighted Fidelity under the same reconstruction view. RankingSHAP evaluates a related but distinct notion through Kendall-similarity preservation and deletion checks. These metrics assess reconstruction of ranker behavior rather than semantic coherence or human usefulness.

Defining how an attribution-induced ranking is evaluated, however, does not resolve what should count as its perturbable feature. RankingSHAP assumes predefined query–document features, whereas RankSHAP uses stemmed-word features shared across documents. However, neither addresses how to combine contextual granularity with shared cross-document feature identity for dense retrieval. We retain the listwise Shapley formulation while introducing a feature space aligned with the representational granularity of dense retrieval.

Semantic Representations in Text Embedding Models.

Text embedding models encode the semantics of sentences or passages into dense vectors rather than scoring documents as independent word occurrences. Sentence-BERT (Reimers and Gurevych, 2019) introduced a siamese architecture for sentence-level similarity, and later models such as E5 (Wang et al., 2024) and GTE (Li et al., 2023) advanced passage-level representation through large-scale contrastive learning, while DPR (Karpukhin et al., 2020) established the bi-encoder retrieval paradigm. Because such systems—including RAG pipelines—often segment documents into passage-level chunks before encoding (Lewis et al., 2020; Guu et al., 2020), a word-level explainer attributes a dense ranking with units finer and more lexical than the ranker’s evidence, whereas raw chunks are document-specific and provide no shared features for listwise per-

Feature unit	Scoring alignment	Coalition space
Word	Directly tied to lexical matching, but weakly aligned with dense semantics	Large sparse vocabulary
Raw chunk	Preserves passage-level context used by dense encoders	Document-specific variables; grows with document length

Table 1: Trade-offs of word and raw chunk feature units for listwise ranking explanation.

turbation. ChunkGroupSHAP closes this gap by grouping semantically related chunks into reusable cross-document features.

3 Methodology

3.1 Preliminaries

Let R denote a retrieval model that assigns a score $s_R(q, d)$ to a query-document pair. For dense rankers, an encoder $E(\cdot)$ maps text to a normalized vector and the score is the cosine similarity $s_R(q, d) = \cos(E(q), E(d))$; for sparse rankers such as BM25, s_R is the lexical retrieval score. Given a query q , the ranker produces a top- n candidate list $D_q = (d_1, \dots, d_n)$ sorted by $s_R(q, d)$.

We explain this ranked list rather than an isolated query-document pair. For a coalition of features $\mathcal{S}$, the ranker is evaluated on the perturbed list and the coalition value is a listlevel utility $v_q(\mathcal{S})$, such as NDCG or Kendall’s τ , comparing the perturbed ranking with the original. Shapley values thus quantify each feature’s marginal contribution to the ranked list, not to an individual document score.

3.2 Feature Units Design

The feature unit determines both the semantic granularity of an explanation and the size of the Shapley coalition space. Table 1 summarizes two considerations for choosing such units: alignment with the ranker’s scoring representation and reuse across the ranked list. For lexical rankers such as BM25, word-level features are directly tied to the scoring mechanism through query-term matches and term statistics. In contrast, dense rankers compute scores from continuous text embeddings, so their scores are not naturally decomposed into independent word occurrences.

Word features therefore provide a clear lexical interpretation, but they can also create a large and

sparse feature space. Moreover, when used for dense rankers, removing isolated words may not correspond well to the semantic units encoded in the embedding representation.

Raw chunks provide a coarser unit of evidence and preserve local context. If document d_i is split into L_i chunks, the top- n list contains $\sum_{i=1}^n L_i$ raw chunk features. However, a raw chunk is typically identified by its document-specific span. Thus, even when two documents contain semantically similar evidence, their chunks correspond to separate coalition variables, which increases the effective coalition space.

Semantic chunk grouping addresses this limitation by mapping semantically similar chunks to shared group identifiers. The resulting groups retain passage-level evidence while making the feature space explicit and bounded by the number of groups k . This allows coalitions to operate over reusable semantic evidence rather than over isolated words or document-specific chunks.

3.3 Semantic Chunk Grouping

ChunkGroupSHAP builds feature units by clustering embedded text chunks. Let $\mathcal{U}$ denote the document universe from which groups are built; the choice of $\mathcal{U}$ defines the grouping scope. In the *corpus-level* variant, $\mathcal{U}$ is the full corpus (or a large candidate union for very large collections), producing reusable group identifiers across queries. In the *query-local* variant, $\mathcal{U} = D_q$, producing groups tailored to a single query. Corpus-level grouping prioritizes reuse across queries, while query-local grouping prioritizes local ranking fidelity.

Chunking and embedding. Each document $d \in \mathcal{U}$ is segmented into overlapping chunks by a tokenizer-based sliding window with window size w and stride s . Let $\mathcal{C} = \bigcup_{d \in \mathcal{U}} \{c_1^{(d)}, c_2^{(d)}, \dots\}$ denote all chunks in $\mathcal{U}$. Each $c \in \mathcal{C}$ is encoded by a grouping embedding model E_g :

$$\begin{aligned} \{c_1^{(d)}, c_2^{(d)}, \dots\} &= \text{CHUNK}(d; w, s) \quad \forall d \in \mathcal{U}, \\ \mathbf{e}_c &= E_g(c) \quad \forall c \in \mathcal{C}. \end{aligned} \tag{1}$$

The grouping model may be the same as the ranker, but need not be.

Clustering. We apply k -means to the chunk embeddings, assigning each chunk to one of k seman-

tic groups $\mathcal{G} = \{g_1, \dots, g_k\}$:

$$k\text{-means}(\{\mathbf{e}_c \mid c \in \mathcal{C}\}, k) \rightarrow g(c) \in \mathcal{G}, \quad (2)$$

$$\forall c \in \mathcal{C}.$$

where $g(c)$ is the group assignment of c . Because a single group identifier can occur in multiple documents, a coalition can perturb shared semantic evidence across the ranked list.

3.4 Group-level Value Function

We define the coalition value function for a fixed query q and candidate list D_q . Its active feature set contains the groups that occur in at least one candidate document:

$$\mathcal{G}_q := \{g(c) \mid d_i \in D_q, c \in \text{chunks}(d_i)\} \subseteq \mathcal{G}. \quad (3)$$

A coalition $\mathcal{S} \subseteq \mathcal{G}_q$ specifies which of these groups remain active.

Document reconstruction. Given $\mathcal{S}$, each d_i is reconstructed from $\text{chunks}(d_i) = \text{CHUNK}(d_i; w, s)$ by retaining only the chunks in active groups, as an ordered subsequence in their original order:

$$\tilde{d}_i(\mathcal{S}) = \text{Concat}([c \in \text{chunks}(d_i) \mid g(c) \in \mathcal{S}]). \quad (4)$$

Overlapping tokens are removed when adjacent retained chunks are joined. If no chunk remains, d_i is treated as empty and receives a zero score.

Scoring. Reconstructed documents are re-scored by the original ranker. For a dense ranker, the query embedding is fixed and each document is re-encoded as $\text{score}_i(\mathcal{S}) = \cos(E(q), E(\tilde{d}_i(\mathcal{S})))$; for BM25, documents are tokenized and scored with the original BM25 parameters.

Coalition Configuration. Let $\sigma_{\mathcal{S}}$ be the ranking induced by $\{\text{score}_i(\mathcal{S})\}_{i=1}^n$. The value measures preservation of the original ranker rather than external qrels: we take the labels $\mathbf{y}^R$ to be the original ranker scores on D_q and, given a cutoff $K \leq n$, compute $v(\mathcal{S}) = \text{NDCG@}K(\sigma_{\mathcal{S}}, \mathbf{y}^R)$. We fix $v(\mathcal{G}_q) = 1$, since the full coalition is the unperturbed reference ranking, and $v(\emptyset) = 0$: under the empty coalition all documents are empty and tie, so a stable sort would preserve the original order and yield an artificially high NDCG. RankingSHAP uses the same framework but replaces NDCG with Kendall’s τ between the original and coalition-induced orders.

3.5 Shapley Value Estimation

We estimate group attributions with KernelSHAP (Lundberg and Lee, 2017), which samples N_{samples} non-empty, non-full coalitions, evaluates $v(\mathcal{S})$ for each, and fits a SHAP-kernel weighted linear regression with a small ridge regularizer. The attribution for group $g \in \mathcal{G}_q$ is ϕ_g , with a base value $\phi_0 = v(\emptyset)$ for the empty coalition, satisfying the efficiency constraint

$$\begin{aligned} \phi_0 + \sum_{g \in \mathcal{G}_q} \phi_g &= v(\mathcal{G}_q), \\ \sum_{g \in \mathcal{G}_q} \phi_g &= v(\mathcal{G}_q) - v(\emptyset) = 1, \end{aligned} \quad (5)$$

where $v(\mathcal{G}_q) = 1$ and $v(\emptyset) = 0$. As values come from sampled coalitions under regularized regression, the ϕ_g are KernelSHAP approximations rather than exact Shapley values. The active feature dimension is $|\mathcal{G}_q| \leq k$, unlike word or raw chunk feature spaces whose size grows with the query’s candidate documents. This bound makes the sampling budget easier to allocate, though a larger k can increase estimation variance.

3.6 Attribution-induced Ranking

Attributions are aggregated back to documents. For each d_i , let $\mathcal{G}_i$ be the groups occurring in it:

$$\mathcal{G}_i := \{g \in \mathcal{G}_q \mid \exists c \in \text{chunks}(d_i) : g(c) = g\}, \quad (6)$$

and its attribution score is $\text{attr}(d_i) = \sum_{g \in \mathcal{G}_i} \phi_g$. In evaluation, we apply the same aggregation restricted to the top attributed features: documents are scored by the sum of selected attributions they contain, and sorting by this score yields an attribution-induced ranking, which is compared with the original ranker output using Fidelity metrics.

4 Experiments

Our experiments compare word features, raw chunks, and semantic chunk groups for listwise ranking explanations. We evaluate how feature unit, grouping scope, and granularity affect faithfulness across sparse and dense rankers, showing that the most faithful unit is setting-dependent rather than fixed.

4.1 Experimental Setup

Datasets and rankers. We use four main retrieval benchmarks: MS MARCO (Bajaj et al.,

MS MARCO										
Ranker	RankSHAP	RankingSHAP	RankSHAP w/ Raw Chunk	ChunkGroupSHAP ($w = 128$)			ChunkGroupSHAP ($w = 64$)			ChunkGroupSHAP (query-local) $k = 300$
				$k = 200$	$k = 300$	$k = 500$	$k = 200$	$k = 300$	$k = 500$	
E5-small	0.135	0.130	0.389	0.044	0.047	0.057	0.025	0.032	0.044	0.394
E5-base	0.133	0.138	0.378	0.033	0.035	0.055	0.018	0.031	0.041	0.384
BM25	0.424	0.431	0.369	0.139	0.159	0.148	0.137	0.133	0.141	0.379

Dataset	Ranker	RankSHAP	RankingSHAP	RankSHAP w/ Raw Chunk	ChunkGroupSHAP ($w = 128$)			ChunkGroupSHAP ($w = 64$)		
					$k = 200$	$k = 300$	$k = 500$	$k = 200$	$k = 300$	$k = 500$
FinanceBench	E5-small	0.221	0.191	0.411	0.346	0.399	0.421	0.302	0.319	0.356
	E5-base	0.237	0.241	0.425	0.378	0.397	0.433	0.283	0.334	0.376
	E5-large	0.208	0.127	0.411	0.367	0.394	0.422	0.304	0.337	0.374
	BM25	0.489	0.430	0.396	0.416	0.399	0.390	0.326	0.348	0.377
AILACaseDocs	E5-small	0.037	-0.032	0.189	0.192	0.210	0.232	0.181	0.214	0.209
	E5-base	0.048	-0.031	0.231	0.220	0.207	0.228	0.166	0.178	0.203
	E5-large	0.028	-0.058	0.218	0.208	0.233	0.228	0.159	0.184	0.209
	BM25	-0.014	0.066	0.106	0.307	0.254	0.281	0.355	0.333	0.269
FinQA	E5-small	0.209	0.198	0.342	0.313	0.332	0.322	0.275	0.297	0.313
	E5-base	0.232	0.234	0.340	0.325	0.333	0.332	0.296	0.308	0.329
	E5-large	0.220	0.238	0.337	0.312	0.338	0.351	0.276	0.308	0.322
	BM25	0.449	0.480	0.372	0.323	0.329	0.336	0.300	0.312	0.302

Table 2: Main benchmark Fidelity_b by dataset and ranker using top-7 attributed features. RankSHAP and RankingSHAP use word features, while RankSHAP w/ Raw Chunk uses non-overlapping 128-token chunks. ChunkGroupSHAP columns report corpus-level chunk groups for each window size w and number of groups k . ChunkGroupSHAP (query-local) reports query-local grouping with $w = 128$ and $k = 300$. Bold indicates the highest Fidelity_b in each row.

2018), FinanceBench (Islam et al., 2023), AILACaseDocs (Bhattacharya et al., 2020), and FinQA (Chen et al., 2021). MS MARCO Passage is retained as an auxiliary short-passage diagnostic rather than part of the main benchmark table. Appendix A.4 reports corpus sizes and document-length statistics.

We explain three dense rankers, E5-small, E5-base, and E5-large (Wang et al., 2024), and one sparse ranker, BM25. Dense rankers use normalized inner-product retrieval. For MS MARCO and MS MARCO Passage, dense rankers rerank BM25 top-1000 candidates; for the other datasets, they search the full corpus.

Explainers. We compare three listwise explainers, all using the same KernelSHAP (Lundberg and Lee, 2017) solver. **RankSHAP** (Chowdhury et al., 2025) uses stemmed word features with NDCG as the coalition value, and **RankingSHAP** (Heuss et al., 2025) uses the same word features with Kendall’s τ . **ChunkGroupSHAP** uses semantic chunk groups obtained by clustering chunk embeddings, with NDCG. We additionally include **RankSHAP w/ Raw Chunk**, which treats non-overlapping 128-token chunks as features without clustering them into groups. It retains RankSHAP’s candidate list, NDCG value function, KernelSHAP estimator, sampling budget, and evaluation protocol, changing the explanation unit from stemmed words to contextual spans.

Chunk groups. Documents are split into overlapping chunks using $(w, s) \in \{(64, 48), (128, 96)\}$. We use *corpus-level* groups when chunks are clustered once over the preprocessed corpus and reused for all queries, and *query-local* groups when chunks are clustered separately within the current query’s candidate list. Unless otherwise stated, corpus-level groups use $k \in \{200, 300, 500\}$. Chunks are embedded by the target dense ranker; for BM25, which does not define an embedding space, groups are built with E5-base embeddings. Appendix A.5 details the clustering universe and candidate-list scope for these grouping variants.

Main and diagnostic settings. The main benchmark uses top-50 candidate lists, 5,000 KernelSHAP samples, and up to 100 queries per dataset except AILACaseDocs, which has 50 queries in total. The raw-chunk baseline is part of this main benchmark and uses non-overlapping 128-token chunks. ChunkGroupSHAP uses corpus-level groups, with additional query-local groups for MS MARCO, whose web-scale queries induce heterogeneous candidate sets. The separate scope diagnostic in Table 3 compares corpus-level and query-local construction on the same 30 queries for AILACaseDocs, FinanceBench, and FinQA.

Unless otherwise noted, the diagnostic analyses use top-10 lists, NDCG@10, 3,000 samples, and at most 30 queries per configuration. The group-granularity sweep instead uses top-

50 lists, NDCG@50, 5,000 samples, and 30 queries; the sampling-budget sweep uses top-50 lists, NDCG@50, and 30 queries.

Evaluation metric. Following the RankSHAP protocol, we measure *Fidelity*: whether the most significant attributed features can reconstruct the original ranked list. For each query, we retain only the top-7 features, following prior work that limits explanations to a small feature set for human readability (Chowdhury et al., 2025), and score each document by the sum of selected attributions it contains; sorting by this score yields an attribution-induced ranking, compared against the original. Because a small feature subset often assigns identical scores to several documents, ties are common, so we define Fidelity with Kendall’s τ_b , which explicitly accounts for ties; we denote this tie-aware metric $Fidelity_b$. We compare the resulting document rankings, not attribution magnitudes across incompatible feature spaces. $Fidelity_b$ therefore measures ranker-output reconstruction under a common feature budget; it does not by itself measure retrieval quality, semantic coherence, or human interpretability.

4.2 Main Results

Table 2 reports $Fidelity_b$ for the main benchmark settings. Its clearest result is the effect of contextual granularity. Across all 11 dense-ranker cells, replacing RankSHAP’s word features with raw chunks increases $Fidelity_b$, with gains ranging from 0.108 to 0.254. In every one of these cells, this word-to-raw-chunk gain is larger than the subsequent numerical difference between raw chunks and the best chunk-group setting. The dense-ranker improvement is therefore driven primarily by moving from isolated lexical features to contextual spans, directly supporting our feature-unit thesis.

Chunk grouping provides a second, distinct contribution. The best chunk-group column is higher than raw chunks in eight of the 11 dense-ranker cells, while raw chunks are highest for AILACaseDocs/E5-base and FinQA/E5-small and E5-base. These cells establish feature-unit outcomes rather than isolate the effect of semantic clustering: raw chunks are non-overlapping, whereas the grouped variants use overlapping chunks and corpus-level or query-local grouping. We isolate the grouping rule in the matched semantic-versus-random analyses below. ChunkGroupSHAP turns document-specific raw chunks into shared cross-

Dataset	Ranker	Corpus Fid. _b	Query Fid. _b
AILACaseDocs	E5-small	0.2767	0.4543
FinanceBench	E5-small	0.5761	0.6762
FinQA	E5-small	0.4391	0.6166

Table 3: Corpus-level versus query-local chunk grouping outside MS MARCO (E5-small, $w = 128$, $k = 200$, top-10 lists, 30 queries). MS MARCO results appear in Table 2.

document feature identifiers and bounds the coalition dimension by k ; the resulting Fidelity difference varies with the dataset, ranker, and grouping scope. On MS MARCO, for example, query-local groups are only 0.005 and 0.006 above raw chunks for E5-small and E5-base, showing that contextual units account for most of the apparent gain over words.

BM25 supplies the complementary lexical boundary. Word-level explanations are strongest on MS MARCO, FinanceBench, and FinQA, and raw chunks are never the best BM25 unit. AILACaseDocs is the long-document exception, where chunk groups are substantially more faithful. Thus contextual units are consistently effective for the tested dense rankers but are not a universal replacement for word features. We analyze the remaining differences through grouping granularity, feature-unit choice, and grouping scope.

4.3 Diagnostic Analyses

4.3.1 Corpus-Level versus Query-Local Grouping

Table 3 makes the grouping-scope trade-off explicit. Corpus-level groups provide reusable feature identities across queries, whereas query-local groups adapt the feature space to the current ranked list at the cost of per-query clustering. Query-local construction is numerically higher on all datasets.

4.3.2 Feature-Space Boundedness

Boundedness is a structural property distinct from contextual granularity and semantic coherence. If document d_i contributes L_i raw chunks, the document-specific raw-chunk space has $M_{\text{raw}} = \sum_i L_i$ features and grows with the number and length of the retrieved documents. ChunkGroupSHAP maps these spans to shared identifiers, so its regression dimension satisfies $M_{\text{group}} \leq k$. In our implementation, $M_{\text{group}} = k$ for corpus-level runs, while query-local runs use $M_{\text{group}} =$

Dataset	Scope	(w, k)	Chunks $\rightarrow$ groups	Semantic	Random
<i>Matched query-local settings with nontrivial merging</i>					
AILACaseDocs	Query-local	(128, 200)	559 $\rightarrow$ 200	0.4543	0.2906
MS MARCO Passage	Query-local	(32, 20)	$\sim 30 \rightarrow 20$	0.6428	0.5366
FinanceBench	Query-local	(128, 20)	$\sim 66 \rightarrow 20$	0.4325	0.3422
<i>Corpus-level counterexamples at $w = 128, k = 200$</i>					
AILACaseDocs	Corpus	(128, 200)	—	0.2767	0.2749
FinanceBench	Corpus	(128, 200)	—	0.5761	0.6045

Table 4: Semantic-versus-random grouping for E5-small on top-10 candidate lists using 3,000 KernelSHAP samples, 30 queries, and top-7 Fidelity_b. Within each row, grouping scope, chunking, k , candidate lists, estimator, and evaluation are matched.

$\min(k, \# \text{chunks})$. Corpus-level groups additionally preserve their identifiers across queries.

This bound makes the coalition dimension and sampling allocation predictable. In the main setting, $N = 5,000$ samples and $k \leq 500$ give $N/M_{\text{group}} \geq 10$. For comparison, AILACaseDocs raw-chunk runs have mean and maximum dimensions of 1,954.3 and 2,859, corresponding to sample-to-feature ratios of 2.56 and 1.75. These ratios characterize the regression design; they do not establish that grouping must improve Fidelity or runtime. With a fixed N , grouping reduces the regression dimension and coalition representation size, but not the number of perturbed-ranking evaluations. Appendix A.1 reports feature counts, and Appendix B.4 shows that dense re-encoding can dominate end-to-end runtime.

4.3.3 Semantic Grouping under Matched Bounds

Table 4 isolates the grouping rule from scope and granularity. In all three tested query-local settings where k forces multiple chunks to merge, semantic grouping yields higher Fidelity_b than matched random grouping, by 0.164, 0.106, and 0.090 on AILACaseDocs, MS MARCO Passage, and FinanceBench, respectively. This supports a specific result: semantic grouping improves rank reconstruction in the tested query-local settings with nontrivial merging.

The corpus-level rows establish the boundary of that result. Semantic and random grouping are nearly tied on AILACaseDocs, while random grouping is higher on FinanceBench. Semantic grouping therefore does not universally increase Fidelity_b across grouping scopes. These counterexamples also reinforce the metric’s limited role: Fidelity_b measures reconstruction of the ranker output, not semantic coherence. We examine granularity and embedding-space coherence as sepa-

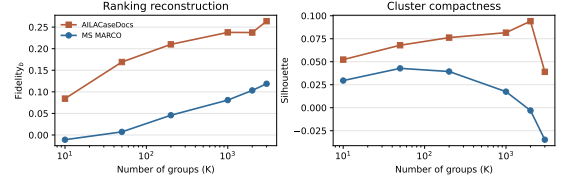


Figure 3: Group-granularity diagnostic for E5-small ChunkGroupSHAP ($w = 128$) using top-50 lists, an NDCG@50 value function, 5,000 KernelSHAP samples, and 30 queries. Left: Fidelity_b as the number of groups k changes. Right: cosine-distance silhouette coefficient.

rate axes below and in Appendix A.2, which also verifies the AILACaseDocs semantic-over-random pattern at three additional group bounds.

4.3.4 Feature Granularity

Figure 3 isolates granularity. Larger k generally improves Fidelity_b on AILACaseDocs (0.084 at $k = 10$ to 0.264 at $k = 3000$) and MS MARCO (−0.011 to 0.119). The silhouette trend differs: compactness peaks at intermediate k and then declines, becoming negative for MS MARCO at large k . Thus cluster compactness is a useful diagnostic but not a proxy for ranking fidelity; finer groups can give SHAP a better reconstruction basis even when the embedding geometry is less compact.

Figure 4 in the Appendix varies the contextual span size from $w = 32$ to $w = 256$. Larger windows help AILACaseDocs and FinanceBench, with the strongest scores at $w = 256$. MS MARCO is more model-dependent: E5-small peaks near $w = 64$ –128, while E5-base improves again at $w = 256$. This sweep is a granularity sensitivity analysis, not a test of semantic grouping: changing w changes both the evidence retained in each span and the number of chunks available for grouping.

4.4 Human Evaluation

We additionally conduct a human reranking study comparing how accurately users *recover* the target ranking from word-level and contextual-chunk explanations. The study is conducted with seven voluntary participants and uses the 30 MS MARCO Passage queries for which E5-small top-10 diagnostic runs are available. For each query, we display five passages drawn from target ranks {1, 3, 5, 7, 9}, using stride two to avoid near-tied adjacent passages. The passages are shown in randomized order, and the target ranking is hidden from participants.

The primary comparison uses two explana-

Condition	N	Mean NDCG@5
Word bars (RankSHAP)	30	0.1893
Query-local chunk cards	30	0.3471

Table 5: Observed mean human reranking scores for the primary comparison from seven voluntary participants.

tion conditions: RankSHAP and query-local ChunkGroupSHAP. RankSHAP is shown as a bar chart over the top attributed word features. Query-local ChunkGroupSHAP ($w = 32$, $s = 24$, $k = 200$) is shown as group cards containing representative chunk snippets. It clusters chunks from the current query’s top-10 passages, and participants drag the passages into their predicted target order. Because these candidate lists contain fewer chunks than the requested $k = 200$, the condition approaches one card per raw chunk. The comparison therefore evaluates word-level versus contextual-chunk explanation presentations, not a causal effect of semantic grouping. An additional corpus-level random-group condition and its result are reported separately in Appendix C.1.

We score each submission by NDCG@5 against the E5-small target ranking, using a linear gain $6 - p$ for target position $p \in \{1, \dots, 5\}$, so that the highest-ranked target passage contributes most. This keeps the human study aligned with the listwise reconstruction setting used in the automatic diagnostics. Appendix C.1 reports the participant instructions, recruitment and compensation details, consent procedure, and ethics-review status.

Table 5 reports mean NDCG@5 of 0.1893 for word bars and 0.3471 for query-local chunk cards. The contextual-chunk presentation records a higher observed mean than the word-bar presentation in this small reranking study, motivating further evaluation of chunk-level evidence presentation. As the query-local condition approaches the raw-chunk limit, this result does not isolate an effect of semantic grouping. Semantic-versus-random claims instead rely on the matched automatic comparisons in Table 4.

5 Conclusion

We presented ChunkGroupSHAP and studied the explanation unit as a first-class design choice for listwise ranking explanation. ChunkGroupSHAP maps contextual chunks to shared cross-document features, preserving a common listwise mask while bounding the KernelSHAP regression dimension

by the number of groups. This structural contribution is distinct from whether semantic grouping increases reconstruction Fidelity in a particular setting.

The controlled unit comparison identifies contextual granularity as the main source of the dense-ranker gains: raw chunks outperform word RankSHAP in all 11 tested E5-family settings, and the word-to-raw-chunk increase is larger than the subsequent difference between raw chunks and the best chunk-group setting in every case. The best chunk-group configuration exceeds raw chunks in eight of those settings. Matched diagnostics further show higher Fidelity for semantic than random grouping in three tested query-local settings with nontrivial merging, but mixed results for corpus-level grouping. Word features remain strongest in three of four BM25 settings.

Together, these results support a precise, setting-dependent conclusion: contextual spans better reconstruct the tested dense bi-encoder rankings, whereas semantic grouping supplies shared and bounded listwise features whose additional Fidelity benefit depends on the corpus, scope, and granularity. Bounding the feature space by k decouples the regression dimension from document length, making coalition representation and sampling allocation predictable as candidate documents grow. Fidelity measures ranker-output reconstruction under a fixed feature budget; semantic coherence and human usefulness remain separate evaluation targets. ChunkGroupSHAP provides a principled tool for analyzing dense-ranking behavior with context-aligned explanation units and a controllable listwise feature space.

Ethical considerations

ChunkGroupSHAP is a research tool for analyzing and auditing ranking behavior. It does not provide causal guarantees about why a ranker produced a decision, nor does it substitute for human review in high-stakes settings: our fidelity metrics measure how well attribution features reconstruct ranker behavior, not fairness, causal validity, or human interpretability.

We use only publicly available datasets, models, and software (MS MARCO, FinanceBench, AILACaseDocs, FinQA, the E5 embedding models, BM25, FAISS, and SHAP-related tooling) for research, consistent with their intended benchmarking use; users of any released code should follow

the original licenses and terms. These corpora may contain names or other sensitive text already present in the source documents. We do not identify individuals, infer private attributes, or release raw text beyond short illustrative excerpts, and released outputs should preserve the access conditions of the source datasets.

We additionally conduct a low-risk human reranking study with voluntary adult participants, collecting only task responses and minimal session metadata and no demographic or sensitive information; full protocol and review status are in Appendix C.1.

Finally, ranking explanations can aid debugging and auditing but may also be misused to overstate system reliability or to optimize content for manipulating rankings. We mitigate this by distinguishing explanation fidelity from human interpretability, reporting limitations, and noting that our method explains behavior within a fixed candidate list rather than a full retrieval pipeline.

Limitations

First, our automatic evaluation targets fidelity to the original ranker, not human interpretability. As the random-grouping diagnostic shows, high fidelity can arise even when a grouping is not semantically meaningful. Our human reranking study (Section 4.4) gives initial evidence on usefulness, but is small in scale, and a larger study is needed to validate chunk-group explanations directly.

Second, ChunkGroupSHAP depends on grouping quality. The chunk window, stride, number of groups, scope, and clustering model all shape the feature space, and we do not provide an automatic procedure for selecting these for a new corpus. In particular, corpus-level grouping can be too coarse for heterogeneous web retrieval, while query-local grouping improves fidelity at added per-query cost.

Third, the method explains rankings within a fixed candidate list. This matches the listwise setup of the compared explainers but does not capture how perturbations would affect full-corpus retrieval. Extending chunk-group attribution to end-to-end pipelines remains future work.

Finally, the attributions are KernelSHAP approximations. The sampling analyses indicate the main results are not driven by insufficient samples, but larger feature spaces and finer groupings can still raise variance and cost, so more efficient or ranker-specific estimators may be needed for very large or

interactive settings.

References

- Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. [Ms marco: A human generated machine reading comprehension dataset](#). *Preprint*, arXiv:1611.09268.
- Paheli Bhattacharya, Kripabandhu Ghosh, Saptarshi Ghosh, Arindam Pal, Parth Mehta, Arnab Bhattacharya, and Prasenjit Majumder. 2020. [Aila 2019 precedent & statute retrieval task](#).
- Zhiyu Chen, Wenhui Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. 2021. [FinQA: A dataset of numerical reasoning over financial data](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3697–3711, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Jaekel Choi, Jungin Choi, and Wonjong Rhee. 2020. [Interpreting neural ranking models using grad-cam](#). *Preprint*, arXiv:2005.05768.
- Tanya Chowdhury, Razieh Rahimi, and James Allan. 2023. [Rank-lime: Local model-agnostic feature attribution for learning to rank](#). In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR '23*, page 33–37, New York, NY, USA. Association for Computing Machinery.
- Tanya Chowdhury, Yair Zick, and James Allan. 2025. [RankSHAP: Shapley value based feature attributions for learning to rank](#). In *The Thirteenth International Conference on Learning Representations*.
- Zeon Trevor Fernando, Jaspreet Singh, and Avishek Anand. 2019. [A study on the interpretability of neural retrieval models using deepshap](#). In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '19*, page 1005–1008. ACM.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. [SimCSE: Simple contrastive learning of sentence embeddings](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. [Realm: retrieval-augmented language model pre-training](#). In *Proceedings of the 37th International Conference on Machine Learning, ICML'20*. JMLR.org.

- Maria Heuss, Maarten de Rijke, and Avishek Anand. 2025. [Rankingshap - faithful listwise feature attribution explanations for ranking models](#). In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, page 381–391, New York, NY, USA. Association for Computing Machinery.
- Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. 2023. [Financebench: A new benchmark for financial question answering](#). *Preprint*, arXiv:2311.11944.
- 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.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. [Retrieval-augmented generation for knowledge-intensive nlp tasks](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.
- Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. [Towards general text embeddings with multi-stage contrastive learning](#). *Preprint*, arXiv:2308.03281.
- Scott M Lundberg and Su-In Lee. 2017. [A unified approach to interpreting model predictions](#). In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Lijun Lyu and Avishek Anand. 2023. [Listwise explanations for ranking models using multiple explainers](#). In *Advances in Information Retrieval*, pages 653–668. Springer Nature Switzerland.
- Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. [MTEB: Massive text embedding benchmark](#). In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence embeddings using Siamese BERT-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. ["why should i trust you?": Explaining the predictions of any classifier](#). In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA. Association for Computing Machinery.
- Stephen Robertson and Hugo Zaragoza. 2009. [The probabilistic relevance framework: Bm25 and beyond](#). *Found. Trends Inf. Retr.*, 3(4):333–389.
- Sourav Saha, Harsh Agarwal, Venkatesh V, Avishek Anand, Swastik Mohanty, Debapriyo Majumdar, and Mandar Mitra. 2025. [ir_explain: A python library of explainable ir methods](#). In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, page 3563–3572, New York, NY, USA. Association for Computing Machinery.
- Lloyd S. Shapley. 1953. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317.
- Jaspreet Singh and Avishek Anand. 2019. [Exs: Explainable search using local model agnostic interpretability](#). In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, page 770–773, New York, NY, USA. Association for Computing Machinery.
- Jaspreet Singh and Avishek Anand. 2020. [Model agnostic interpretability of rankers via intent modelling](#). In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 618–628, New York, NY, USA. Association for Computing Machinery.
- Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. [Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models](#). *Preprint*, arXiv:2104.08663.
- Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024. [Text embeddings by weakly-supervised contrastive pre-training](#). *Preprint*, arXiv:2212.03533.
- Puxuan Yu, Razieh Rahimi, and James Allan. 2022. [Towards explainable search results: A listwise explanation generator](#). In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 669–680, New York, NY, USA. Association for Computing Machinery.

Appendix

A Experimental Details

A.1 Feature-Space Size and Group Coverage

Table 6 reports the number of features exposed to the SHAP regression for each query. Word features are unique normalized word features from the query and its top-50 retrieved documents. Raw chunk features are non-overlapping tokenizer chunks used by RankSHAP w/ Raw Chunk; we use a 128-token chunk for all document datasets and a 32-token chunk for MS MARCO Passage because passage documents are much shorter. The resulting feature space is substantially larger than the number of documents being explained, especially for long-document datasets such as AILACaseDocs.

Table 7 reports candidate-list group coverage, not the configured group bound or corpus-level regression dimension. For corpus-level grouping, the corpus is partitioned into k groups, but a query’s retrieved documents contain chunks from only a subset of them. Because the clustering is refit for each k , this covered subset generally grows as the partition becomes finer. Query-local grouping instead clusters only the current candidate documents, so every realized group is covered and the count is $\min(k, \# \text{chunks})$. Values below are the mean and maximum numbers of distinct covered groups across queries.

A.2 Bound Robustness and Embedding-Space Coherence

Table 8 separates two properties of grouping. Panel (a) varies the group bound while keeping the AILACaseDocs query-local protocol fixed. Because each query contains approximately 559 chunks, all three settings perform nontrivial merging. Semantic grouping records higher Fidelity_b than matched random grouping at every tested bound. This shows that the query-local semantic-over-random result is not specific to the $k = 200$ setting reported in the main text.

Panel (b) measures embedding-space coherence in the corpus-level settings where Fidelity_b is tied or favors random grouping. For each group containing at least two chunks, we compute pairwise cosine similarity using the normalized E5-small chunk embeddings and pool the within-group chunk pairs. The reported difference is the semantic-group mean minus the matched random-group mean. Semantic groups are more coherent

by 0.058 on AILACaseDocs and 0.085 on FinanceBench even though their Fidelity outcomes are mixed. Coherence and rank reconstruction are therefore distinct empirical axes. Because the same embedding space is used for clustering and this diagnostic, the result is a geometry sanity check, not independent evidence of human interpretability.

A.3 Ranker Specifications

Table 9 summarizes the retrieval backends used in the experiments. We use the term ranker consistently for both dense embedding retrievers and the sparse BM25 baseline. The E5-family rankers share a 512-token maximum input length. BM25 is not parameterized by neural weights and scores the lexical document representation with $k_1 = 1.5$ and $b = 0.75$.

Table 10 reports qrel-based retrieval quality for the same rankers. We compute NDCG@10, NDCG@25, and NDCG@50 from the saved top-50 rankings using the standard exponential gain ($2^{\text{rel}} - 1$). This retrieval-quality NDCG is separate from the self-ranking linear-gain NDCG used inside the explanation value function.

A.4 Dataset Statistics

Table 11 summarizes the corpus size, query count, and qrel count for the datasets used in our experiments. The benchmark suite spans web documents and passages, financial reports, legal case documents, and financial QA evidence. We include MS MARCO Passage in the same summary because it is used for the short-passage chunk-window analysis.

Table 12 reports document length statistics. Lengths are measured in word counts using alphanumeric tokenization. The query count is included for scale; the length statistics are computed over corpus documents. The datasets differ substantially in document granularity: MS MARCO Passage consists of short passages, while FinanceBench and FinQA contain short to medium evidence documents, and AILACaseDocs contains long-form legal reports.

A.5 Implementation and Package Settings

The implementation uses Python 3.12 with NumPy, SciPy, pandas, sentence-transformers, scikit-learn, FAISS, NLTK, and SHAP-compatible weighted-regression tooling. Dense retrieval indexes use FAISS IndexFlatIP over L2-normalized embeddings, so inner product is equivalent to cosine simi-

Dataset	#Queries	Word Mean	Word Max	Raw Chunk Mean	Raw Chunk Max
MS MARCO	100	6,498.2	14,595	687.1	2,633
MS MARCO Passage	100	649.9	1,087	144.2	189
FinanceBench	100	1,826.5	2,370	264.6	333
AILACaseDocs	50	9,694.0	12,025	1,954.3	2,859
FinQA	100	3,279.6	3,952	360.9	412

Table 6: Query-level word and raw-chunk feature-space sizes. Raw chunks use a 128-token window except MS MARCO Passage, where we use a 32-token window.

Dataset	Scope	Window	#Docs	$k = 200$	$k = 300$	$k = 500$
MS MARCO	Corpus	128	50	35.6 / 86	41.7 / 92	51.1 / 122
MS MARCO	Query-local	128	50	—	300.0 / 300	—
FinanceBench	Corpus	128	50	104.4 / 143	141.1 / 193	232.5 / 307
FinanceBench	Query-local	128	10	66.1 / 152	—	—
AILACaseDocs	Corpus	128	50	133.1 / 160	177.9 / 217	257.7 / 327
AILACaseDocs	Query-local	128	10	200.0 / 200	—	—
FinQA	Corpus	128	50	98.4 / 117	125.8 / 147	156.2 / 187
FinQA	Query-local	128	10	89.0 / 112	—	—

Table 7: Candidate-list coverage of ChunkGroupSHAP groups by grouping option. Entries show the mean/maximum number of distinct group identifiers represented in a query’s retrieved documents.

larity. E5-family models are run through sentence-transformers with normalized embeddings and the standard query: and passage: input prefixes. BM25 is implemented in pure Python rather than through rank-bm25; all BM25 experiments use $k_1 = 1.5$ and $b = 0.75$.

Text features are extracted with a deterministic lowercased alphanumeric token regular expression. For dense rankers, masking removes matching raw-text token spans while preserving the remaining punctuation, casing, and whitespace; for BM25, masking removes matching token identities before sparse scoring. ChunkGroupSHAP forms chunks with the window and stride settings reported in Section 4.1; overlapping chunks are deduplicated when perturbed documents are reconstructed. Corpus-level chunk groups are produced with scikit-learn k -means using a fixed random seed of 42. For large chunk matrices with at least 50,000 rows we use mini-batch k -means with batch size $\max(4096, 20k)$ and three initializations; otherwise we use standard k -means with automatic initialization selection. KernelSHAP coalition sampling also uses seed 42 unless otherwise specified.

B Additional Analyses

B.1 Visible-Token Coverage under E5 Truncation

Dense E5 scores are computed by encoding query: and passage: inputs. For E5-small, E5-base, and E5-large, the tokenizer-visible input is limited to

512 tokens. Therefore, for any scoring call, the dense ranker can only attend to the prefix that remains visible after the perturbed text is reconstructed and truncated. A word or chunk-group feature outside this visible prefix cannot directly affect the embedding for that scoring call; it can only matter if perturbing earlier text changes which content moves into the visible prefix.

Table 13 audits this constraint on completed E5-small outputs. We measure whether each query’s top-7 features by absolute SHAP value have at least one occurrence in the original 512-token visible prefix of the query or retrieved documents. The audit shows that semantic chunk-group features are highly prefix-supported on the financial, legal, and QA datasets, while word-level features are less stable on very long legal documents.

B.2 Cross-Document Feature Support

Because listwise attribution masks the same feature across all documents in a ranked list, a useful group-level feature should appear in more than one document. We therefore measure how widely each chunk group is distributed within the top-50 documents for each query. For query q and group g , let $D_q(g)$ be the set of top-50 documents that contain at least one chunk assigned to g . We report the number of active groups, the average and median value of $|D_q(g)|$, the fraction of active groups with

(a) Query-local bound robustness on AILACaseDocs

k	Chunks $\rightarrow$ groups	Semantic	Random
50	$\sim 559 \rightarrow 50$	0.388	0.164
100	$\sim 559 \rightarrow 100$	0.408	0.232
150	$\sim 559 \rightarrow 150$	0.378	0.243

(b) Corpus-level Fidelity and coherence

Dataset	Sem. Fid. _b	Rand. Fid. _b	Δ cosine
AILACaseDocs	0.2767	0.2749	+0.058
FinanceBench	0.5761	0.6045	+0.085

Table 8: Robustness and coherence diagnostics for E5-small. Panel (a) uses query-local groups with $w = 128$, top-10 candidate lists, 3,000 KernelSHAP samples, 30 queries, and top-7 Fidelity_b; semantic and random conditions are matched within each row. Panel (b) uses corpus-level $w = 128$, $s = 96$, and $k = 200$. Δ cosine is semantic minus random pooled within-group pairwise cosine similarity.

Ranker	Type	Params	Dim.	Layers	Max Tokens	Pooling / Scoring
E5-small	Dense encoder	33.4M	384	12	512	Mean pooling
E5-base	Dense encoder	109.5M	768	12	512	Mean pooling
E5-large	Dense encoder	335.1M	1,024	24	512	Mean pooling
BM25	Sparse lexical	—	—	—	Full text	$k_1 = 1.5, b = 0.75$

Table 9: Ranker specifications used in the experiments. Parameter counts are computed from local model weights.

$|D_q(g)| \geq 2$, and normalized document entropy:

$$H_q(g) = -\frac{1}{\log 50} \sum_{d \in D_q(g)} p(d | g) \log p(d | g),$$

where $p(d | g)$ is the fraction of chunks in group g that occur in document d within the top-50 list. Higher coverage and entropy indicate that the feature better satisfies the cross-document support condition required for listwise perturbation.

B.3 Chunk-Window and Sampling-Budget Sweeps

Figure 4 reports Fidelity_b across chunk window sizes for E5-small and E5-base at $k = 200$. On AILACaseDocs and FinanceBench, larger windows generally improve reconstruction, with the strongest scores at $w = 256$, indicating that longer domain documents benefit from more local context per chunk. MS MARCO is more model-dependent: E5-small peaks near $w = 64$ –128 while E5-base improves again at $w = 256$, reflecting the heterogeneous web setting’s sensitivity to the interaction between window size and grouping scope.

We vary the KernelSHAP sampling budget for E5-small (Figure 5). On FinanceBench and AILACaseDocs, ChunkGroupSHAP stays above both word-level baselines across budgets, especially at $k = 500$, with the clearest gain from 1k to 3k samples; on MS MARCO the order reverses, consistent with corpus-level groups being too coarse there. Sampling budget thus matters but does not override the choice of feature space and grouping granularity.

B.4 Runtime and Computational Cost

Table 15 reports observed wall-clock time per query for the main E5-small top-50 setting. These numbers should be interpreted as implementation runtime rather than a monotonic function of feature count. All methods use a fixed number of KernelSHAP samples, so reducing the number of features lowers the regression dimension and attribution output size, but it does not by itself reduce the number of perturbed ranking evaluations. For E5, the dominant cost is repeatedly reconstructing perturbed top-50 documents and encoding them with the dense ranker. In the current implementation, ChunkGroupSHAP reconstructs documents from overlapping chunks for every coalition, which can make it comparable to or slower than word masking even though it uses fewer features.

Compute infrastructure. Experiments were run on a mixture of local and cloud GPU machines, including AWS G5 instances with NVIDIA A10G GPUs and local workstations equipped with NVIDIA RTX 3090 and RTX 5090 GPUs. Dense ranker inference and chunk embedding used GPU acceleration; BM25 indexing and scoring used CPU execution. We report observed per-query wall-clock runtime in Table 15. The total GPU-hour budget was not centrally tracked because experiments were run in several batches across these machines, but the main benchmark used the fixed settings described in Section 4.1: top-50 candidate lists and 5,000 KernelSHAP samples per query.

Dataset	Ranker	NDCG@10	NDCG@25	NDCG@50
MS MARCO	E5-small	0.3789	0.4214	0.4328
	E5-base	0.3895	0.4289	0.4447
	E5-large	0.3981	0.4313	0.4454
	BM25	0.2915	0.3054	0.3289
MS MARCO Passage	E5-small	0.4440	0.4607	0.4625
	E5-base	0.4097	0.4265	0.4355
	E5-large	0.4281	0.4452	0.4484
	BM25	0.2111	0.2347	0.2479
FinanceBench	E5-small	0.6407	0.6520	0.6560
	E5-base	0.6402	0.6569	0.6674
	E5-large	0.6828	0.6944	0.7021
	BM25	0.3185	0.3702	0.3948
AILACaseDocs	E5-small	0.1704	0.2040	0.2464
	E5-base	0.1691	0.1967	0.2363
	E5-large	0.1876	0.2228	0.2733
	BM25	0.1671	0.1946	0.2675
FinQA	E5-small	0.3934	0.4159	0.4362
	E5-base	0.4474	0.4780	0.4929
	E5-large	0.4358	0.4601	0.4746
	BM25	0.7651	0.7737	0.7794

Table 10: Qrel-based retrieval quality by dataset and ranker. Bold indicates the best ranker for each dataset and cutoff.

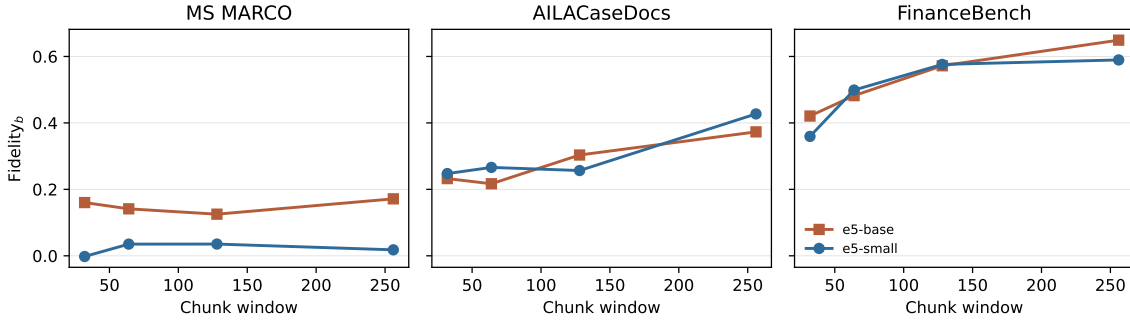


Figure 4: Chunk-window diagnostic for E5-small and E5-base ChunkGroupSHAP with $k = 200$. Fidelity_b is computed from the top-7 attributed groups on top-10 ranking lists with 30 evaluated queries per setting.

Dataset	#Docs	#Queries	#Qrels
MS MARCO	3,213,835	5,193	5,193
MS MARCO Passage	8,841,823	6,980	7,437
FinanceBench	145	150	150
AILACaseDocs	186	50	50
FinQA	380	1,138	1,138

Table 11: Dataset-level corpus, query, and relevance-judgment statistics.

C Human Evaluation Details

C.1 Human Evaluation Protocol

The human evaluation in Section 4.4 is a low-risk reranking study designed to test whether different explanation units help participants recover a hidden target ranking. The task uses only public MS MARCO Passage texts and E5-small rankings. Seven voluntary adult participants completed the study. Participants were recruited through an internal volunteer pool and were not paid. We did not

collect demographic attributes or sensitive personal information.

Before starting the study, participants were informed that the task was part of a research evaluation of ranking explanations, that participation was voluntary, and that their submitted rankings and interaction metadata would be used for aggregate analysis. The participant-facing task instruction shown on each task page was:

Use the information on the left to correctly order the documents on the right.

Each task displayed a query, one explanation panel, and five passage cards in a randomized initial order. Participants dragged the passage cards into their predicted target order and submitted the ordering. The target order, model scores, qrels, raw document identifiers, query identifiers, and SHAP internals were hidden from participants. Explanation panels used neutral presentation labels: RankSHAP

Dataset	#Docs	#Queries	Mean	Median	P25	P75	P90	P99	Max
MS MARCO	3,213,835	5,193	1,173	601	303	1,162	2,380	10,352	336,402
MS MARCO Passage	8,841,823	6,980	58	51	43	67	94	130	287
FinanceBench	145	150	266	231	99	339	571	921	1,049
AILACaseDocs	186	50	4,720	3,340	1,841	5,416	8,714	33,375	40,518
FinQA	380	1,138	593	588	471	727	864	1,055	1,345

Table 12: Document length distribution across all datasets. Lengths are word counts under alphanumeric tokenization.

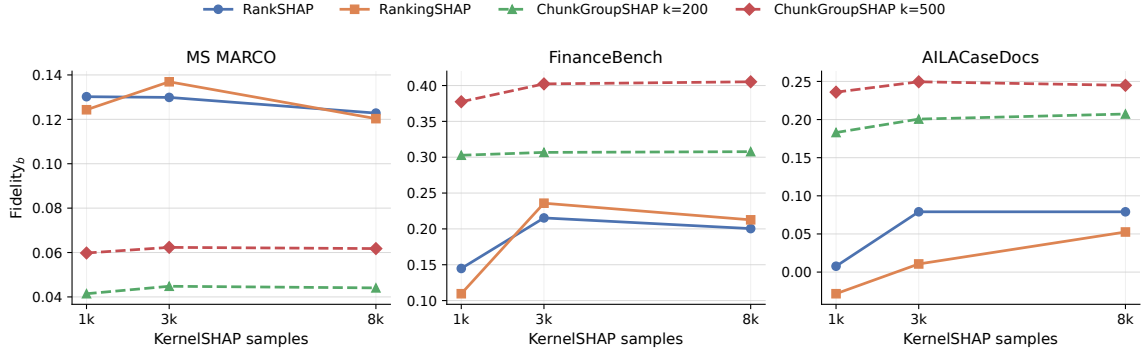


Figure 5: Effect of the KernelSHAP sampling budget for E5-small. Each panel compares word-level RankSHAP and RankingSHAP against ChunkGroupSHAP with $w = 128$ and $k \in \{200, 500\}$ on top-50 lists with an NDCG@50 value function and 30 queries. All values report Fidelity_b using the top-7 attributed features.

Dataset	Word	ChunkGroup
MS MARCO	71.7%	74.7%
MS MARCO Passage	100.0%	100.0%
FinanceBench	98.7%	99.1%
AILACaseDocs	58.3%	96.3%
FinQA	91.6%	98.7%

Table 13: Visible-prefix support of top-7 E5-small features. Word features are checked against the 512-token query/document prefix; ChunkGroup features are checked by whether the group has at least one chunk occurrence whose chunk start falls within the visible prefix of a retrieved document.

was shown as important terms, while ChunkGroupSHAP conditions were shown as important passage groups. The study compared three conditions: RankSHAP, query-local ChunkGroupSHAP, and ChunkGroupSHAP with random grouping. Both chunk-card conditions used $w = 32$, $s = 24$, and $k = 200$; the former clustered the current query’s top-10 passages, while the latter used corpus-level random group assignments. The main text reports the primary comparison between RankSHAP and query-local ChunkGroupSHAP. For completeness, the additional corpus-level random-group condition recorded a mean NDCG@5 of 0.2995 over the same 30 queries. Because it differs from the query-local condition in both grouping rule and scope, we do not treat their difference as evidence for a causal

effect of semantic grouping. Figure 6 shows the web interface used for the human evaluation.

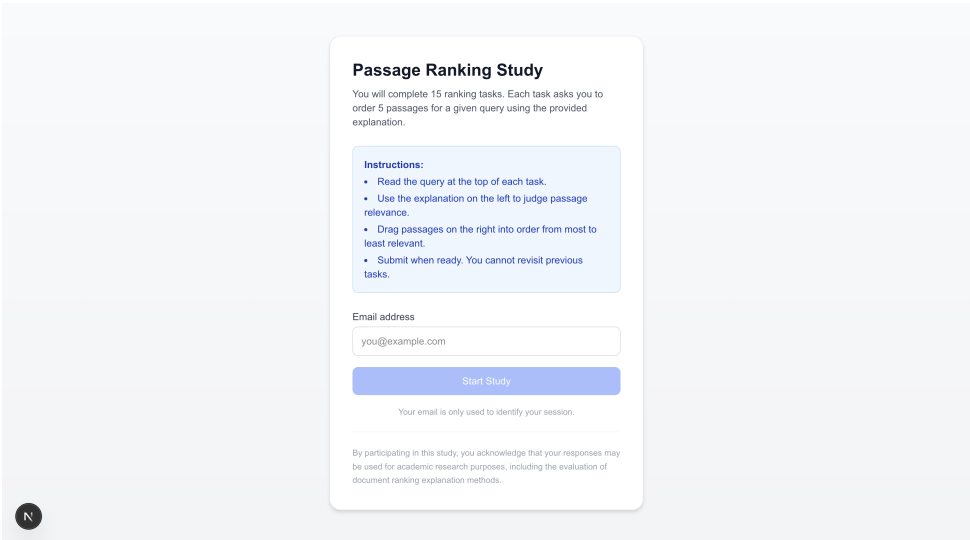
The evaluation dataset contains 30 MS MARCO Passage queries. For each query, the five displayed passages correspond to E5-small target ranks $\{1, 3, 5, 7, 9\}$, and the hidden ideal order is the E5-small order over those five passages. The analysis records the submitted order, condition, elapsed time, and interaction count for each response. Human reranking quality is reported as mean NDCG@5 by condition using the linear target-position gains defined in Section 4.4.

No formal ethics review board approval or exempt determination was obtained. We treat the study as low risk because it uses public benchmark passages, collects only reranking responses and minimal session metadata, does not ask for demographic or sensitive personal information, and reports only aggregate results.

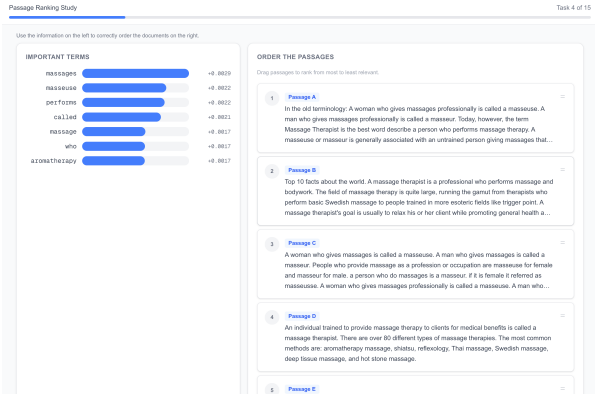
D Reproducibility, Ethics, and Artifact Use

D.1 Reporting Conventions

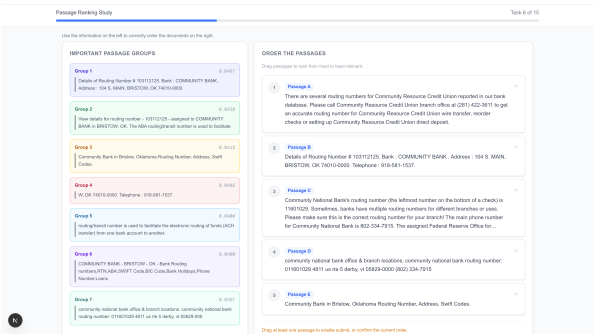
Unless otherwise stated, reported Fidelity values are averages over evaluated queries under the setting named in the corresponding table or figure caption. Tables report the number of evaluated



(a) Home screen.



(b) Word-feature reranking screen.



(c) Chunk-group-feature reranking screen.

Figure 6: Human evaluation web application. The top panel shows the home screen for the reranking study; the bottom panels show the task interface with word-feature explanations and chunk-group-feature explanations.

Dataset	Active Groups	Docs/Group	Median Docs/Group	Shared Groups	Entropy
MS MARCO	35.56	5.36	2.09	59.71%	0.218
MS MARCO Passage	4.72	19.93	16.50	71.55%	0.500
FinanceBench	104.35	2.53	1.82	57.43%	0.164
AILACaseDocs	133.10	6.06	3.87	74.44%	0.281
FinQA	98.39	2.50	1.82	54.52%	0.150

Table 14: Cross-document support of chunk-group features within top-50 ranked lists. Statistics are averaged over queries using E5-small retrieval and corpus-level E5-small chunk groups with window 128, stride 96, and $k = 200$. Shared Groups denotes the fraction of active groups that occur in at least two top-50 documents.

Dataset	Word RankSHAP	Word RankingSHAP	ChunkGroupSHAP
MS MARCO	946.1	941.9	1,113.7
MS MARCO Passage	328.9	325.4	366.9
FinanceBench	639.6	640.0	650.8
AILACaseDocs	1,634.6	1,628.7	1,823.3
FinQA	703.8	699.0	727.9

Table 15: Observed average elapsed seconds per query for E5-small top-50 experiments. ChunkGroupSHAP uses corpus-level $k = 500$ groups.

queries when the comparison uses a diagnostic subset or query intersection. Feature-space size and candidate-list group-coverage tables report mean and maximum counts across queries. Runtime values are observed wall-clock seconds per query averaged over the completed E5-small top-50 runs. For hyperparameter analyses, we report the swept values directly rather than selecting a single tuned configuration on a separate validation set. The main benchmark table reports the configured grid for chunk window size and group count, while the diagnostic figures and tables report additional sweeps over group count, chunk window, grouping scope, random grouping, and KernelSHAP sampling budget.

D.2 Artifact Use, Access Conditions, and Data Handling

Table 16 summarizes the main scientific artifacts used in the experiments. We use these artifacts only for research on retrieval, ranking, and explanation. We do not redistribute the original corpora, model weights, or third-party software artifacts as part of this work; users of any released code or derived outputs must obtain the underlying artifacts from their original sources and comply with the corresponding licenses, data-use agreements, and model terms. For generated outputs, we retain only attribution scores, aggregate metrics, feature identifiers, and limited qualitative snippets needed to illustrate the method.

The datasets are public retrieval benchmarks, but they may contain names of people, organizations,

legal parties, financial entities, URLs, or other sensitive text that is already present in the original sources. For benchmark corpora, we did not collect private user data, infer private attributes, or perform additional identity linking. Separately, the human evaluation records only voluntary reranking responses and minimal session metadata, as described in Appendix C.1. As a data-protection step, the experimental pipeline operates on local copies of the benchmark corpora and reports aggregate metrics and attribution outputs rather than republishing the full underlying documents. Qualitative examples should be limited to short excerpts necessary for understanding the explanation and should preserve the access conditions of the source dataset.

D.3 Use of AI Writing Assistance

AI assistants were used as writing-support and editing tools during manuscript preparation. The authors used them to help draft, revise, and polish text. The authors reviewed and are responsible for all final manuscript content, technical claims, experimental descriptions, and reported results.

Artifact	Type	Use in this work	Access / derivative handling
MS MARCO, MS MARCO Passage	Dataset	Retrieval benchmark and candidate lists	Public benchmark; no corpus redistribution
FinanceBench	Dataset	Financial retrieval benchmark	Public benchmark; no corpus redistribution
AILACaseDocs	Dataset	Legal retrieval benchmark	Public benchmark; no corpus redistribution
FinQA	Dataset	Financial QA evidence retrieval benchmark	Public benchmark; no corpus redistribution
E5-family encoders	Models	Dense retrieval and chunk embedding	Used through released model checkpoints/terms
BM25 implementation	Ranker	Sparse lexical retrieval baseline	In-repository implementation; no third-party BM25 package
FAISS, scikit-learn, NumPy/SciPy	Software	Indexing, clustering, numerical computation	Used as software dependencies under their licenses

Table 16: Scientific artifacts and their use in the experiments. We use all artifacts for research and benchmarking only and do not redistribute source corpora or model weights.