

Beyond Nearest Neighbors: Semantic Compression and Graph-Augmented Retrieval for Enhanced Vector Search

Rahul Raja^{1 2 3} Arpita Vats^{4 2}

Abstract

Vector databases typically rely on approximate nearest neighbor (ANN) search to retrieve the top- k closest vectors to a query in embedding space. While effective, this approach often yields semantically redundant results, missing the diversity and contextual richness required by applications such as retrieval-augmented generation (RAG), multi-hop QA, and memory-augmented agents. We introduce a new retrieval paradigm: *semantic compression*, which aims to select a compact, representative set of vectors that captures the broader semantic structure around a query. We formalize this objective using principles from submodular optimization and information geometry, and show that it generalizes traditional top- k retrieval by prioritizing coverage and diversity. To operationalize this idea, we propose *graph-augmented vector retrieval*, which overlays semantic graphs (e.g., kNN or knowledge-based links) atop vector spaces to enable multi-hop, context-aware search. We theoretically analyze the limitations of proximity-based retrieval under high-dimensional concentration and highlight how graph structures can improve semantic coverage. Our work outlines a foundation for meaning-centric vector search systems, emphasizing hybrid indexing, diversity-aware querying, and structured semantic retrieval. We make our implementation publicly available to foster future research in this area⁵.

Work does not relate to position at LinkedIn. ¹Carnegie Mellon University, Pittsburgh, USA ²LinkedIn, California, USA ³Stanford University, Palo Alto, USA ⁴Boston University, Boston, USA. Correspondence to: Rahul Raja <rauhl.110392@gmail.com>, Arpita Vats <arpita.vats09@gmail.com>.

Proceedings of the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

⁵https://github.com/rauhlrj/icml_vecdb_experiments

1. Introduction

Vector search systems, often powered by approximate nearest neighbor (ANN) algorithms (Johnson et al., 2019), are widely used for tasks such as semantic search (Guo et al., 2020), recommendation (Wang et al., 2021), and retrieval-augmented generation (RAG) (Lewis et al., 2020). These systems typically retrieve the top- k vectors closest to a query in embedding space. While effective for ensuring local relevance, this approach frequently results in semantically redundant outputs that lack contextual diversity—an important limitation for multi-hop QA, fact-based summarization, and memory-augmented reasoning (Asai & Hajishirzi, 2020).

We propose a new retrieval paradigm: *semantic compression*, which aims to return a compact and representative set of vectors that captures the broader semantic structure around a query. Unlike conventional top- k retrieval that prioritizes proximity, semantic compression emphasizes both relevance and diversity by framing retrieval as a set selection problem. We formalize this objective using submodular optimization (Krause & Golovin, 2014) and information geometry, showing it to be a generalization of proximity-based methods.

To make semantic compression practical, we introduce **graph-augmented vector retrieval**, where symbolic edges (e.g., kNN, clustering, or knowledge-based links) are added to embedding spaces (Liu et al., 2021). These graphs enable context-aware, multi-hop retrieval methods such as personalized PageRank (PPR) (Haveliwala, 2003), allowing the discovery of semantically diverse but non-local results. Our evaluations demonstrate that graph-based methods, especially with dense symbolic connections, outperform pure ANN retrieval in semantic diversity while maintaining high relevance.

The main contributions of this paper are:

- We introduce **semantic compression**, a retrieval objective focused on selecting diverse and representative results beyond local proximity.
- We formalize semantic compression using submodular optimization and demonstrate how it generalizes standard top- k retrieval.

- We propose **graph-augmented retrieval** by enriching vector spaces with symbolic edges to support multi-hop, context-aware search.
- We conduct extensive experiments on both semantic compression and graph-based retrieval, showing that these methods significantly improve semantic diversity while maintaining high relevance.

2. Related Work

Approximate Nearest Neighbor (ANN) Search

ANN methods form the backbone of modern vector databases, enabling fast retrieval in high-dimensional spaces (Muja & Lowe, 2014; Johnson et al., 2019). Classical approaches include locality-sensitive hashing (LSH) (Indyk & Motwani, 1998), inverted file indexing with product quantization (IVF-PQ) (Jégou et al., 2011), and graph-based structures such as HNSW (Malkov & Yashunin, 2018). While these techniques provide low-latency top- k retrieval, they are primarily designed to minimize geometric distance and do not explicitly account for semantic diversity or coverage.

Diversity-Promoting Retrieval

Several works in information retrieval have explored promoting diversity in ranking, notably through maximal marginal relevance (MMR) (Carbonell & Goldstein, 1998), determinantal point processes (DPPs) (Kulesza & Taskar, 2012), and submodular selection (Krause & Golovin, 2014). However, these approaches are often applied as post-processing steps over retrieved candidates, rather than being integrated into the retrieval mechanism itself. Our work differs by embedding diversity into the retrieval objective and infrastructure directly, motivated by semantic compression.

Graph-Augmented Retrieval

Incorporating structure into retrieval systems via graphs has gained interest in recent years. Graph-based ANN methods like HNSW (Malkov & Yashunin, 2018) implicitly exploit proximity graphs, while knowledge graphs (Cui & et al., 2019) and co-occurrence networks (Asai & Hajishirzi, 2020) have been used to guide multi-hop reasoning. Our proposal builds on this idea by explicitly augmenting vector space with semantic graphs that support multi-hop, context-aware search beyond simple distance metrics.

Retrieval for Language Models

Retrieval-augmented generation (RAG) (Lewis et al., 2020) and similar architectures rely heavily on vector search to provide factual grounding to LLMs. However, these systems often retrieve semantically redundant passages, leading to hallucinations or missed evidence (Shi et al., 2023). Our method offers a structured retrieval alternative that emphasizes representational coverage, potentially improving downstream performance in LLM pipelines.

Comparison to Our Work

While prior research addresses individual components—fast retrieval, diversity, or graph-based augmentation—our work provides a unified framework for semantic compression and graph-augmented retrieval. We contribute theoretical foundations, objective formulations, and architectural implications for next-generation vector search systems.

3. Semantic Compression

3.1. Motivation

Traditional vector retrieval retrieves the top- k vectors nearest to a query point q based on a similarity function (e.g., cosine or dot product) (Johnson et al., 2019; Muja & Lowe, 2014). However, this often results in semantically redundant items—vectors that are close in space but convey overlapping information (Ash & et al., 2021). This is especially problematic in tasks where diversity and broad coverage of related concepts are essential, such as open-domain question answering (Asai & Hajishirzi, 2020) or document summarization (Carbonell & Goldstein, 1998; Xie & et al., 2022).

To address this limitation, we introduce the notion of *semantic compression*: the task of selecting a small set of vectors that captures the most semantically informative, diverse, and representative content from the neighborhood of a query.

3.2. Problem Definition

Let $\mathcal{V} = \{v_1, v_2, \dots, v_n\} \subset \mathbb{R}^d$ denote a set of candidate vectors retrieved by an initial ANN pass around a query vector $q \in \mathbb{R}^d$, and let $S \subseteq \mathcal{V}$ be a subset of size k to be returned to the downstream model or user. The goal of semantic compression is to construct a subset S that is both representative of the semantic content near q and internally diverse.

We formalize this by the following objective:

$$\max_{S \subseteq \mathcal{V}, |S|=k} \underbrace{\sum_{v \in \mathcal{V}} \max_{s \in S} \text{sim}(v, s)}_{\text{Coverage}} + \lambda \cdot \underbrace{\sum_{\substack{u, v \in S \\ u \neq v}} (1 - \text{sim}(u, v))}_{\text{Diversity}}$$

where:

- $\text{sim}(u, v)$ is a similarity measure (e.g., cosine similarity) between vectors u and v .
- The **coverage term** encourages selection of vectors that "cover" the semantic space of \mathcal{V} by ensuring each point $v \in \mathcal{V}$ is similar to at least one item in S .
- The **diversity term** penalizes selection of semantically similar items, promoting representation of different subregions or concepts.

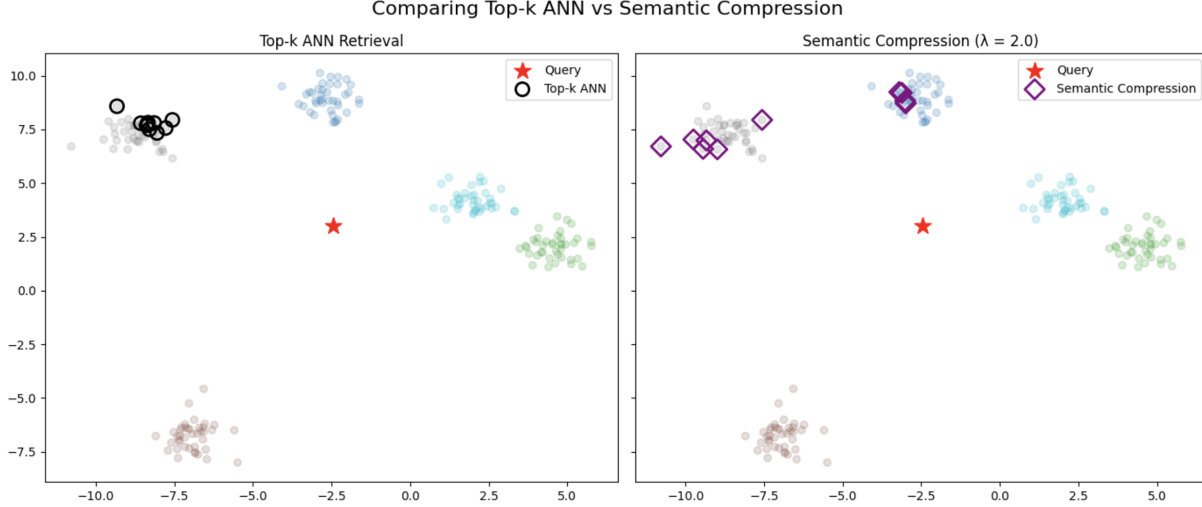


Figure 1. Comparison of Top- k ANN (left) and Semantic Compression (right) in 2D. The query is marked in red. While ANN retrieves points from a single dense region, semantic compression surfaces points across multiple clusters.

- $\lambda \geq 0$ balances the trade-off between semantic fidelity and representational spread.

3.3. Submodular Formulations and Optimization

The semantic compression problem can be cast as a submodular maximization task. Submodular functions exhibit diminishing returns, making them amenable to efficient greedy approximations. Let the utility function $f(S)$ be:

$$f(S) = \sum_{v \in \mathcal{V}} \max_{s \in S} \text{sim}(v, s) + \lambda \cdot \sum_{\substack{u, v \in S \\ u \neq v}} (1 - \text{sim}(u, v))$$

The first term behaves like a facility-location function, measuring how well S represents \mathcal{V} . The second term behaves like a diversity-regularizer based on pairwise distances in the similarity space.

To optimize this objective, a greedy algorithm can be employed:

- Initialize $S \leftarrow \emptyset$.
- Iteratively add the element $v^* \in \mathcal{V} \setminus S$ that maximizes the marginal gain: $f(S \cup \{v^*\}) - f(S)$.
- Repeat until $|S| = k$.

This greedy method achieves a $(1 - 1/e)$ -approximation guarantee for monotonic submodular functions, and is efficient for real-world deployment.

3.4. Connection to Retrieval Systems

This formulation subsumes standard top- k ANN retrieval as a special case. When $\lambda = 0$, the objective reduces to selecting the k vectors with highest similarity to the query, equivalent to nearest neighbor retrieval. As λ increases, the selection balances relevance with diversity, promoting coverage across semantically distinct regions. This tradeoff allows the retrieval process to surface results that span multiple subtopics or latent dimensions, rather than concentrating in a single high-density cluster.

3.5. Implementation of Semantic Compression in Practice

Semantic compression can be deployed as a lightweight second-stage reranking module following an approximate nearest neighbor (ANN) retrieval step. Below, we describe a concrete implementation pipeline that integrates seamlessly into existing vector retrieval systems.

Step 1: Candidate Generation via ANN. Given a query vector $q \in \mathbb{R}^d$, retrieve an initial candidate pool of size N using an ANN engine (e.g., FAISS, HNSW):

$$\mathcal{V} = \{v_1, v_2, \dots, v_N\}, \quad v_i \in \mathbb{R}^d$$

Step 2: Similarity Matrix Computation. Construct:

- A query-to-candidate similarity vector:

$$s_q = [\text{sim}(q, v_1), \dots, \text{sim}(q, v_N)] \in \mathbb{R}^N$$

- A full candidate-to-candidate similarity matrix:

$$S = VV^T \in \mathbb{R}^{N \times N}$$

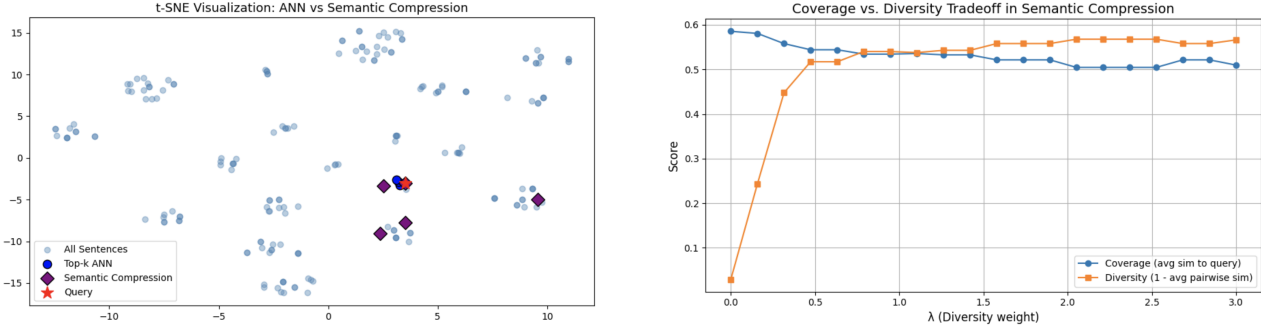


Figure 2. **Left:** t-SNE projection of sentence embeddings, comparing Top- k ANN (blue) with Semantic Compression (purple diamonds) for a single query (red star). While ANN retrieves a tightly clustered set, semantic compression surfaces points from multiple semantic regions. **Right:** Tradeoff between coverage (similarity to query) and diversity (pairwise dissimilarity) as the diversity weight λ increases. As λ grows, the selection becomes more diverse with only a modest drop in query relevance.

where $V \in \mathbb{R}^{N \times d}$ contains row-stacked normalized vectors. Cosine similarity can be used as the scoring function.

Step 3: Subset Selection via Greedy Optimization. Using the utility function defined in Section 3.3 (Submodular Formulation), apply the greedy algorithm to select a compressed subset $S \subset \mathcal{V}$, of size k , that maximizes semantic coverage and diversity.

- **Performance:** For typical values $N = 100$, $k = 10$, greedy selection takes less than 1 ms on modern hardware.
- **Batching:** Matrix operations ($S = VV^\top$) and masked indexing enable efficient batching across queries on GPU.
- **Modularity:** This reranking layer is model-agnostic and plug-and-play, requiring no retraining.
- **Extensibility:** Additional relevance scores (e.g., from a graph-based propagation model) can be blended with similarity matrices for hybrid reranking.

This implementation strategy makes semantic compression practical for real-time inference settings, enabling improved retrieval without changes to the core ANN infrastructure.

3.6. Experiment: Comparing ANN and Semantic Compression

To evaluate the effectiveness of semantic compression, we design a controlled experiment using synthetically generated data. A set of 200 two-dimensional vectors is sampled from five well-separated clusters, simulating distinct semantic regions. A query vector is constructed as the mean of one representative point from each cluster, representing a composite concept that spans multiple topics.

We compute cosine similarity between the query and all candidate vectors and extract the top-50 most similar points to form a candidate retrieval pool. From this pool, we compare two selection strategies for identifying a subset of $k = 10$ vectors: (1) standard top- k retrieval based solely on similarity, and (2) semantic compression that balances similarity with diversity across clusters.

Figure 1 shows the retrieved vectors under both strategies. While the top- k method tends to concentrate on a single cluster, semantic compression produces a more diverse set spanning all five clusters. This demonstrates the model’s ability to capture broader semantic coverage without sacrificing relevance.

To analyze the tradeoff between relevance and diversity in semantic compression, we present two complementary visualizations: a t-SNE projection of the embedding space and a coverage-diversity tradeoff plot.

In Figure 2 (left), Top- k ANN retrieves points densely clustered near the query, often resulting in redundant selections. Semantic compression, on the other hand, produces a more spatially distributed set that captures multiple semantic modes. This leads to improved representational coverage, which is beneficial for downstream tasks such as reranking or retrieval-augmented generation.

Figure 2 (right) illustrates how the method balances similarity and diversity as the diversity weight λ increases. When $\lambda = 0$, the method reduces to standard nearest neighbor retrieval. As λ increases, diversity rises steadily while coverage remains relatively stable.

These visualizations highlight semantic compression as a tunable, model-agnostic approach that enables flexible control over semantic coverage in retrieval settings.

4. Graph-Augmented Vector Retrieval

4.1. Motivation

While semantic compression improves retrieval diversity, it is fundamentally limited by the local geometry of the embedding space (Poerner et al., 2020). Real-world semantic relations—such as synonymy, hierarchy, and multi-hop associations—are often non-metric and cannot be fully captured by vector similarity alone (Nickel & Kiela, 2017; Asai & Hajishirzi, 2020; Bolukbasi & et al., 2016).

To address this, we propose *graph-augmented vector retrieval*, which overlays a semantic graph on top of the embedding space. The graph encodes latent or symbolic relationships between items, enabling retrieval paths that incorporate global structure, contextual dependencies, and long-range reasoning. This hybrid formulation bridges local similarity with higher-order semantic organization.

4.2. Graph Construction and Traversal

We construct a semantic graph $G = (\mathcal{V}, \mathcal{E})$, where nodes $\mathcal{V} = \{v_1, v_2, \dots, v_n\} \subset \mathbb{R}^d$ correspond to embedded items (e.g., documents or entities), and edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ encode semantic or structural relationships. This graph augments the embedding space with contextual or symbolic structure.

Edge Formation. Edges in G can be defined via:

- **kNN-based similarity:** Connect each node to its k nearest neighbors under cosine similarity:

$$\text{sim}(v_i, v_j) = \frac{v_i^\top v_j}{\|v_i\| \cdot \|v_j\|}.$$

- **External relations:** Use hyperlinks, co-occurrence, or citation edges from metadata or interaction logs.
- **Symbolic graphs:** Incorporate curated relationships from knowledge graphs (e.g., ConceptNet, Wikidata).

Graph Traversal. To propagate relevance from a query-seeded set $\mathcal{V}_q \subset \mathcal{V}$, we perform semantic expansion via graph-based methods:

- **Random walks:** Sample bounded-length walks from \mathcal{V}_q , accumulating visited nodes.
- **Personalized PageRank:**

$$\mathbf{r} = \alpha \cdot \mathbf{s} + (1 - \alpha) \cdot \mathbf{A}^\top \mathbf{r},$$

where \mathbf{A} is the normalized adjacency matrix and \mathbf{s} is the query seed vector.

- **Graph neural networks:** Iteratively update node embeddings via:

$$\mathbf{h}_v^{(l+1)} = \text{AGG} \left(\left\{ \mathbf{h}_u^{(l)} : (u, v) \in \mathcal{E} \right\} \right).$$

This graph-based augmentation introduces multi-hop semantic reasoning and symbolic structure into the retrieval process, addressing limitations of pure distance-based retrieval in complex domains.

4.3. Hybrid Scoring and Retrieval

To bridge the gap between local embedding similarity and global semantic relationships, we introduce a hybrid scoring framework that integrates vector-based relevance with graph-based propagation. Given a semantic graph $G = (\mathcal{V}, \mathcal{E})$, and a query embedding $\mathbf{q} \in \mathbb{R}^d$, we first retrieve a candidate pool $\mathcal{V}_q \subset \mathcal{V}$ using standard ANN search in the vector space. Our objective is to refine the ranking of nodes in $\mathcal{V}_q \cup \mathcal{N}(\mathcal{V}_q)$, where $\mathcal{N}(\cdot)$ denotes neighbors in the graph G , by considering both geometric proximity and graph-induced connectivity.

1. Vector-Based Relevance. The local relevance of a node $v \in \mathcal{V}$ is defined via cosine similarity between its embedding $\mathbf{v} \in \mathbb{R}^d$ and the query vector:

$$S_{\text{vec}}(v, q) = \cos(\mathbf{v}, \mathbf{q}) = \frac{\mathbf{v}^\top \mathbf{q}}{\|\mathbf{v}\| \cdot \|\mathbf{q}\|}.$$

2. Graph-Based Influence. To capture semantic signals that are not evident from local proximity, we compute a relevance diffusion score $S_{\text{graph}}(v, q)$ by propagating query affinity through the graph structure. One effective method is **Personalized PageRank (PPR)**, which computes a stationary distribution over nodes biased towards the query neighborhood:

$$\mathbf{r} = \alpha \cdot \mathbf{s} + (1 - \alpha) \cdot \mathbf{r} \mathbf{A},$$

where \mathbf{s} is a one-hot or soft seed vector centered on \mathcal{V}_q , \mathbf{A} is the normalized adjacency matrix of G , and $\alpha \in (0, 1)$ controls the restart probability. The score $S_{\text{graph}}(v, q) = \mathbf{r}[v]$ then reflects how reachable v is from the query region under random walks.

3. Hybrid Scoring Function. We define the final relevance score for each node $v \in \mathcal{V}_q \cup \mathcal{N}(\mathcal{V}_q)$ as a convex combination of the vector and graph-based components:

$$R(v | q) = (1 - \beta) \cdot S_{\text{vec}}(v, q) + \beta \cdot S_{\text{graph}}(v, q),$$

where $\beta \in [0, 1]$ modulates the influence of structural context. This formulation smoothly interpolates between pure ANN retrieval ($\beta = 0$) and purely graph-based exploration ($\beta = 1$).

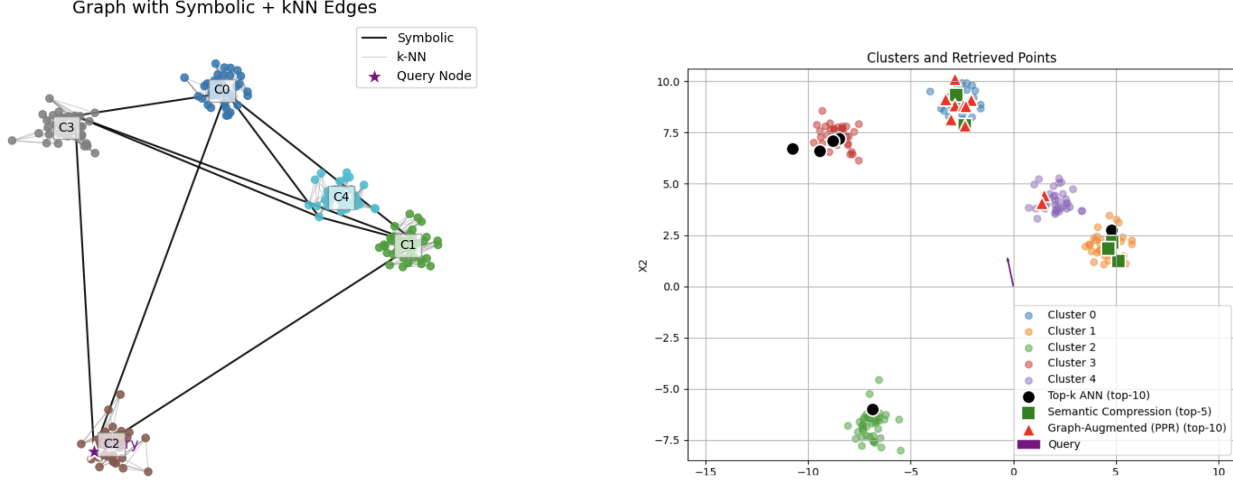


Figure 3. **Left:** Hybrid graph constructed by combining symbolic edges (black) and k -nearest neighbor edges (gray). This enriched structure supports semantic exploration through Personalized PageRank. **Right:** Retrieved points for a fixed query (shown in red) across three strategies: Top- k ANN (blue), Semantic Compression (purple diamonds), and Graph-Augmented Retrieval (green). While ANN focuses on local neighborhoods and Semantic Compression samples from diverse clusters, the graph-based method retrieves highly relevant points concentrated in a semantic region.

The hybrid approach can be viewed as an instance of **multi-view retrieval**, where the vector space and graph topology represent two complementary manifolds. The weighted fusion ensures robustness to deficiencies in either view—for example, embedding collapse or graph sparsity.

To ensure scalability:

- The graph G can be restricted to the kNN subgraph over \mathcal{V}_q , significantly reducing memory and compute.
- S_{graph} can be precomputed for frequent queries or efficiently approximated via truncated random walks.
- The scores $R(v | q)$ support batching and GPU acceleration, making them deployable in production retrieval stacks.

This hybrid retrieval mechanism significantly enhances semantic generalization, particularly in cases where the query concept has no direct match in the corpus but is indirectly connected via ontological or contextual relationships.

4.4. Experiment : Graph-Augmented Retrieval with Sparse Symbolic Cluster Connectivity

We evaluate the three retrieval strategies on a synthetic 2D dataset with clustered structure: (i) Top- k Approximate Nearest Neighbors (ANN), (ii) Semantic Compression using cluster centers, and (iii) Graph-Augmented Retrieval based on Personalized PageRank (PPR) over a symbolic k -NN graph.

To assess retrieval quality, we compute two metrics: **Relevance**, the average cosine similarity between retrieved items and the query, and **Diversity**, defined as one minus the average pairwise cosine similarity among the retrieved items. Each method retrieves the top-10 results for a fixed query.

Method	Relevance \uparrow	Diversity \uparrow
Top- k ANN	0.5174	0.8068
Semantic Compression	0.4798	0.5718
Graph-Augmented (PPR)	0.9688	0.0671

Table 1. Comparison of relevance and diversity across retrieval methods.

Graph-Augmented Retrieval yields the highest relevance by leveraging both symbolic and similarity-based structure. PPR walks tend to remain within semantically consistent regions, which explains the lower diversity. In contrast, Top- k ANN, based on geometric proximity, offers a better balance between relevance and diversity. Semantic Compression provides moderate scores by returning cluster centroids, which capture representative semantics but lose finer granularity.

Figure 3 offers a visual comparison. The left subfigure shows the hybrid symbolic+ k NN graph used by the PPR-based retrieval, while the right subfigure overlays the retrieved points from each strategy on the original clustered space. We observe distinct retrieval patterns for each method due to their inherent mechanisms. Top- k ANN retrieves items tightly around the query because it directly selects the nearest neighbors in the embedding space based purely on

geometric proximity. This approach favors local neighborhoods and can capture fine-grained similarities but may miss semantically diverse points outside the immediate vicinity. In contrast, Semantic Compression samples more broadly across clusters by selecting representative cluster centroids or summaries, thereby promoting diversity by design. However, this broader coverage can reduce relevance since some centroids may lie farther from the query, reflecting a trade-off that favors diversity over strict query similarity. Finally, Graph-Augmented Retrieval using Personalized PageRank (PPR) focuses on a dense region of semantically aligned points because the diffusion of relevance scores over the hybrid symbolic+ k NN graph propagates influence through tightly connected nodes. The graph structure encodes both direct semantic relations (symbolic edges) and local similarity (k NN edges), which results in prioritizing nodes strongly connected to the query. This mechanism leads to high relevance but reduces diversity, as the retrieval concentrates within a semantically cohesive neighborhood.

4.5. Enhancing Graph-Augmented Retrieval via Dense Symbolic Cross-Cluster Connections

We also explore the impact of significantly increasing the number of symbolic edges by connecting cluster heads to semantically similar nodes across different clusters. Unlike the initial approach, which only linked cluster heads to a limited number of other heads, this enhancement establishes symbolic edges whenever the cosine similarity between a cluster head and any node in a different cluster exceeds a threshold (0.85). This creates a denser and more semantically meaningful symbolic graph, enriching the graph structure with additional cross-cluster connections.

Figure 4 illustrates the combined retrieval graph with both k -nearest neighbor (KNN) edges and these expanded symbolic edges. KNN edges are shown in light gray, whereas symbolic edges are highlighted in red dashed lines. Cluster heads are marked as prominent black stars. This visualization clearly shows the enriched symbolic connectivity bridging distant but semantically related points.

Method	Relevance	Diversity
Top-k ANN	0.9987	0.0013
Semantic Compression	0.9987	0.0013
Graph-Augmented(PPR)	0.9168	0.1590

Table 2. Relevance and diversity scores for different retrieval methods with enhanced symbolic edges.

The results reveal a clear trade-off between relevance and diversity. Both Top-k ANN and Semantic Compression methods achieve very high relevance (≈ 0.999) but at the cost of extremely low diversity (≈ 0.0013), indicating their retrieved points are highly focused and similar. In contrast,

Graph-Augmented Retrieval using Personalized PageRank (PPR) exhibits slightly lower relevance (0.9168) but substantially higher diversity (0.1590). This suggests that incorporating expanded symbolic edges enables the graph to better capture semantic relationships across clusters, leading to richer and more diverse retrieval results.

The expanded symbolic connectivity effectively bridges different clusters by linking semantically related points that are not necessarily nearest neighbors in the original vector space. This enriched graph structure facilitates the PPR algorithm to traverse meaningful semantic pathways beyond local neighborhoods, enhancing the diversity of retrieved results without a significant drop in relevance.

In summary, increasing symbolic edges provides a promising direction to balance relevance and diversity in graph-augmented retrieval systems, surpassing the limitations of purely nearest-neighbor based approaches.

5. Conclusion

We presented a new retrieval paradigm that goes beyond traditional top- k nearest neighbor search by prioritizing semantic diversity and representational coverage. Our approach, semantic compression, formalizes retrieval as a submodular optimization problem, enabling selection of compact yet informative results. We further extended this idea with graph-augmented retrieval, integrating symbolic edges into vector space to support multi-hop, context-aware search. Empirical results across diverse retrieval configurations demonstrate that our methods significantly improve semantic diversity without sacrificing relevance. This work lays the foundation for more meaning-aware vector retrieval systems, with applications spanning RAG, question answering, and agent memory retrieval. Future work will explore adaptive graph construction and tighter integration with large language model pipelines.

6. Impact Statement

Our work introduces a retrieval framework that prioritizes semantic diversity and contextual relevance, addressing a core limitation of current vector search systems. By formalizing semantic compression and integrating symbolic graph structures, we enable retrieval methods better aligned with the needs of modern language models and multi-step reasoning tasks. This paradigm has broad implications for the design of future retrieval-augmented systems, including factual grounding, multi-hop question answering, and memory-augmented agents. It encourages the research community to rethink retrieval beyond geometric proximity and move toward meaning-centric information access.

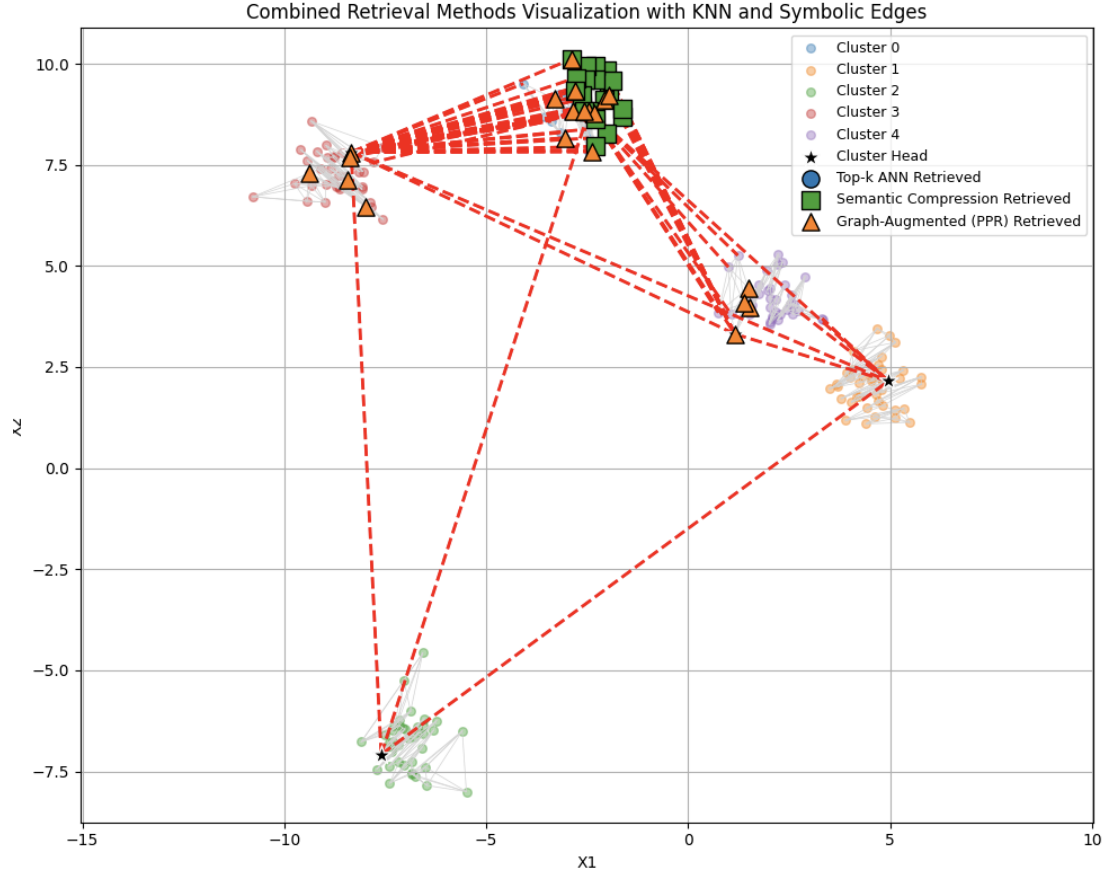


Figure 4. Visualization of combined retrieval methods on the dataset with enhanced symbolic edges. KNN edges are shown in light gray, while symbolic edges are highlighted as red dashed lines. Cluster heads are marked by black stars. Retrieved points from Top-k ANN, Semantic Compression, and Graph-Augmented Retrieval (PPR) are shown with distinct markers and colors.

References

- Asai, A. and Hajishirzi, H. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *ICLR*, 2020.
- Ash, J. and et al. On sampling strategies for neural retrieval. In *NeurIPS*, 2021.
- Bolukbasi, T. and et al. All words are not created equal: Semantic specialization for word representations. *arXiv preprint arXiv:1609.00893*, 2016.
- Carbonell, J. and Goldstein, J. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.
- Cui, L. and et al. Kbqa: A knowledge base question answering benchmark dataset and analysis. In *EMNLP*, 2019.
- Guo, R., Sun, F., Lindgren, E., et al. Accelerating large-scale inference with anisotropic vector quantization. In *Proceedings of Machine Learning and Systems (MLSys)*, 2020.
- Haveliwalla, T. H. Topic-sensitive pagerank. *WWW*, 2003.
- Indyk, P. and Motwani, R. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *STOC*, 1998.
- Jégou, H., Douze, M., and Schmid, C. Product quantization for nearest neighbor search. In *IEEE TPAMI*, 2011.
- Johnson, J., Douze, M., and Jégou, H. Billion-scale similarity search with gpus. In *IEEE Transactions on Big Data*, 2019.
- Krause, A. and Golovin, D. Submodular function maximization. In *Tractability: Practical Approaches to Hard Problems*, 2014.

- Kulesza, A. and Taskar, B. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 2012.
- Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.
- Lewis, P., Perez, E., Piktus, A., et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *NeurIPS*, 2020.
- Liu, H., Weston, J., and Kiela, D. Learning discrete representations via neural clustering. In *ICLR*, 2021.
- Malkov, Y. A. and Yashunin, D. A. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE TPAMI*, 2018.
- Muja, M. and Lowe, D. G. Scalable nearest neighbor algorithms for high dimensional data. *IEEE TPAMI*, 2014.
- Nickel, M. and Kiela, D. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*, 2017.
- Poerner, N., Schütze, H., and Roth, B. Evaluating neural semantic encoders with syntactic tree distances. In *ACL*, 2020.
- Shi, W., Chen, X., and et al. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*, 2023.
- Wang, X., He, X., Wang, M., et al. A survey on neural recommendation: From collaborative filtering to content and context aware recommendation. *ACM Computing Surveys*, 2021.
- Xie, Y. and et al. Unified summarization evaluation with diversity and relevance focused metrics. In *ACL*, 2022.