International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.4 (2025) pp. 8555-8564
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Semantic–Lexical Fusion: Improving Retrieval Accuracy for AI-Driven Knowledge Systems

Karthik Chakravarthy Cheekuri*

Microsoft Technologies, USA
* **Corresponding Author Email:** cheekuri.karthik@gmail.com - **ORCID:** 0000-0002-9947-7850

Article Info:

DOI: 10.22399/ijcesen.4267
Received : 12 September 2025
Revised : 01 November 2025
Accepted : 06 November 2025

Keywords

hybrid retrieval,
semantic search,
lexical matching,
retrieval-augmented generation,
information retrieval systems

Abstract:

Abstract should be about 100-250 words. It should be written times new roman and 10 punto. Large language models increasingly rely on external knowledge retrieval to generate accurate, context-aware responses. Dense vector representations enable semantic similarity matching but struggle with exact term identification, structured metadata constraints, and domain-specific identifiers common in enterprise environments. This framework integrates dense embedding-based retrieval with token-level inverted indexing to address these limitations. The architecture employs dual indexing structures, a unified query decomposition layer, and a weighted scoring mechanism that combines cosine similarity with BM25 relevance signals. Field-level filtering and access control mechanisms ensure compliance with organizational constraints while maintaining semantic generalization.Performance characterization across production-scale deployments demonstrates that hybrid architectures achieve 60-150ms query latency with 1.0-2.0GB memory footprint per million documents, while pure semantic approaches require 768MB-1.5GB and pure lexical systems consume 200-500MB. Index construction analysis reveals vector encoding demands 15-50 hours for million-document collections compared to minutes for inverted indices, though batch processing strategies mitigate operational impact. Sensitivity analysis across weighting parameters identifies optimal semantic-lexical balance at $\alpha = 0.6$, achieving peak F1 score of 0.847 and demonstrating 15-30% accuracy improvements over single-paradigm methods.Evaluation across open-domain corpora (MS MARCO, Natural Questions) and enterprise document collections demonstrates improved precision and recall, particularly for queries containing product identifiers, user entities, and structured filters. The system supports retrieval-augmented generation pipelines, conversational interfaces, and knowledge-grounded chatbots where both conceptual relevance and deterministic matching are essential. Results indicate that fusing complementary retrieval paradigms provides robust performance across diverse query types while maintaining interpretability and production-grade reliability for AI-driven applications. Cost analysis for enterprise deployments reveals 50-80% infrastructure premium for hybrid systems compared to single-approach implementations, justified by reduced hallucinations, improved user satisfaction, and decreased support escalations in operational AI applications.

1. Introduction

1.1 Background and Rationale

Large language models have altered how information retrieval functions within modern computing infrastructures. Users now formulate queries using natural language patterns instead of traditional keyword strings, anticipating responses that reflect semantic comprehension alongside contextual relevance. Dense vector representations constitute core elements in current search frameworks, allowing systems to detect conceptual connections extending past direct term equivalence.

1.2 Operational Constraints of Semantic Retrieval

Industrial environments have experienced notable progress in optimizing semantic retrieval mechanisms [1]. Real-world deployments,

however, uncover meaningful limitations when organizations depend solely on semantic methodologies. Precise term identification presents difficulties when queries reference particular entities like invoice codes, merchandise identifiers, or account numbers. Assessment protocols evaluating semantic retrieval dependability across specialized computing sectors [2] have documented accuracy challenges affecting enterprise implementations where deterministic results are mandatory.

1.3 Principal Obstacles

Multiple core barriers surface when implementing purely semantic frameworks in active deployments. Structured identifiers alongside domain-specific notation require exact retrieval capabilities that semantic methods frequently cannot deliver. Enterprise systems necessitate thorough metadata filtering coupled with authorization governance structures that dense retrieval designs fundamentally lack. Vocabulary fluctuations within specialized fields generate further complications, where terminology disparities and technical language can produce semantic deviation and unsuccessful retrieval transactions.

1.4 Consequences for AI Applications

Retrieval shortcomings ripple through AI-enabled applications relying on accurate contextual data. Platforms utilizing retrieval-augmented generation, conversational knowledge systems, and automated assistants encounter diminished effectiveness when foundational search components yield imprecise or unrelated material. Defective retrieval workflows contribute to manufactured outputs in language model answers, produce mismatched responses, or establish security weaknesses through insufficient permission frameworks, positioning dependable hybrid retrieval as an essential prerequisite for operational AI implementations.

1.5 Identified Gaps and Proposed Solutions

Examinations of semantic retrieval dependability within industrial settings [2] alongside neural retrieval augmentation tactics [1] have moved the discipline forward substantially. Nevertheless, actionable architectures that successfully combine semantic interpretation with deterministic filtering capabilities remain insufficiently developed. This framework presents a consolidated semantic-lexical retrieval apparatus merging dense vector similarity calculations with token-based ranking procedures and structured filtering components. The resulting infrastructure builds a workable basis for developing reliable, deployment-ready search platforms underpinning AI-enabled knowledge applications.

2. Related Work and Background

2.1 Vector-Based Passage Encoding Strategies

Dense passage encoding frameworks coupled with transformer-based sentence models have revolutionized textual representation for similarity assessment. These techniques map queries alongside documents into high-dimensional spaces where geometric proximity signals semantic correspondence. Domain adaptation through meta-learning strategies has strengthened cross-domain transfer in passage retrieval systems, mitigating performance degradation when shifting between different knowledge areas. Cosine angular distance typically quantifies relevance between embedded representations.

Multiple operational barriers constrain purely vector-based retrieval effectiveness. Embedding models encounter difficulties capturing subtle meaning variations, especially for uncommon vocabulary items lacking adequate training instances. Structured attribute filtering poses challenges since vector encodings do not inherently preserve categorical metadata. Performance optimization at scale has motivated approximate nearest neighbor implementations like FAISS and HNSW graph structures, trading marginal accuracy losses for substantial computational efficiency gains across extensive document repositories.

Production deployments reveal critical latency characteristics that shape architectural decisions. Vector similarity operations using approximate nearest neighbor algorithms typically achieve sub-50ms query latencies for collections up to 10 million passages when properly tuned, though exact performance depends heavily on dimensionality, index parameters, and hardware specifications. This responsiveness makes semantic search viable for interactive applications requiring sub-200ms total response times.

Index construction presents more substantial computational demands. Building dense vector indices requires encoding all documents through transformer models—computationally intensive operations consuming 50-200ms per document depending on text length and model complexity. For a 1 million document collection, initial vector encoding might require 15-50 hours on standard server hardware, though batch processing and GPU acceleration can reduce this substantially. This extended build time influences deployment

strategies, particularly when rapid system initialization is necessary.

Incremental updates reveal important operational constraints distinguishing vector from traditional approaches. Adding documents to vector indices requires encoding new content and inserting vectors into approximate nearest neighbor structures, operations taking 100-300ms per document including embedding generation. For systems processing continuous document streams, this latency influences architecture decisions around batch update frequencies and index refresh strategies. Organizations frequently adopt scheduled batch updates rather than real-time indexing to manage computational overhead.

Memory overhead constitutes another critical production consideration. Dense vector representations typically consume 768-1536 bytes per document for common embedding dimensions (192-384 dimensions with 32-bit floats), yielding 768MB-1.5GB memory requirements per million documents purely for vector storage. Approximate nearest neighbor graph structures add 50-100% overhead for connectivity information, effectively doubling base requirements. These memory demands directly impact infrastructure costs for large-scale deployments spanning tens of millions of documents.

2.2 Statistical Term Frequency Methods

Established information retrieval builds upon term occurrence statistics through TF-IDF weighting schemes and BM25 probabilistic scoring. These computational approaches determine relevance by examining how terminology distributes throughout document corpora. Production search engines including Lucene and Elasticsearch have implemented these methodologies for operational use, demonstrating reliability across diverse domains and scales.

Statistical techniques deliver notable benefits such as result explainability, where individual scoring factors remain transparent and auditable, plus accuracy when literal term presence matters. Performance characteristics prove favorable for production environments. Inverted index lookups through BM25 scoring generally complete within 20-100ms for collections approaching 10 million documents, with performance scaling logarithmically through term posting list optimizations. This consistent latency advantage over vector-based approaches makes lexical methods particularly suitable for high-throughput scenarios.

Index construction efficiency represents a significant operational advantage. Inverted indices typically process 1000-5000 documents per second, completing million-document collections in minutes rather than hours. This efficiency stems from straightforward tokenization and posting list construction, avoiding computationally intensive neural encoding. Organizations can initialize or rebuild lexical indices with minimal downtime, supporting agile deployment practices and rapid iteration cycles.

Incremental updates to inverted indices occur nearly instantaneously through append operations on posting lists. This characteristic makes lexical approaches ideal for high-velocity document streams where continuous updates are essential. Real-time indexing adds negligible overhead, typically under 10ms per document, enabling systems to reflect content changes within seconds of ingestion.

Memory requirements demonstrate favorable characteristics compared to vector approaches. Inverted indices typically require 200-500 bytes per document for term posting information, translating to 200-500MB per million documents. This represents roughly one-third the memory footprint of dense vector representations, making lexical systems more economical for large-scale deployments where infrastructure costs significantly impact total cost of ownership.

However, fundamental weaknesses exist. Terminology misalignment creates retrieval gaps when identical concepts appear with distinct vocabulary across queries and target content, producing failures despite conceptual alignment. These frameworks cannot identify semantic equivalence between rephrased statements or alternative expressions without manual expansion dictionaries, limiting their effectiveness for natural language queries where paraphrasing is common.

2.3 Combined Retrieval Frameworks

Modern retrieval platforms increasingly merge diverse ranking signals via integrated architectures. Late interaction designs alongside learned sparse encoding represent recent advances in unified retrieval construction. Hybrid systems incorporating generative reordering stages illustrate how sequential processing balances initial recall against final precision targets. Staged ranking workflows commonly begin with rapid candidate identification using efficient approximation techniques, then apply resource-intensive models to reduced result sets during subsequent reordering phases.Hybrid architectures introduce additional latency overhead through parallel retrieval coordination, score normalization procedures, and result fusion operations. Well-optimized

implementations add approximately 10-30ms of processing time beyond the slower of the two component retrievers, yielding total query response times in the 60-150ms range for medium-scale deployments. This latency profile remains acceptable for interactive applications where sub-200ms response times meet user experience requirements. The marginal overhead proves worthwhile when accuracy improvements directly impact business outcomes, such as in customer support systems, internal knowledge bases, or retrieval-augmented generation pipelines.The dual-index approach requires provisioning for both structures simultaneously, roughly doubling memory footprint compared to single-paradigm approaches. For a million-document collection, hybrid systems typically consume 1.0-2.0GB of memory, though compression techniques can mitigate this by 30-40%. Production deployments must balance this increased resource consumption against improved retrieval quality and operational flexibility. Organizations processing 10,000+ queries per second face significant computational infrastructure requirements, where the marginal hybrid overhead translates to measurable cost increases.Maintenance complexity represents another operational consideration. Hybrid systems must coordinate updates across both index types while maintaining consistency. The asymmetry between instant inverted index updates and slower vector encoding creates opportunities for optimization through batched vector refresh strategies. Systems commonly update inverted indices in real-time while processing vector updates in scheduled batches—hourly or daily intervals—accepting temporary staleness in semantic matching to maintain operational efficiency. This mixed-mode update strategy balances freshness requirements against computational resource constraints.Cost analysis for typical enterprise deployments reveals the infrastructure implications. For a system managing 10 million documents processing 1000 queries per second, pure vector implementations might require $800-1200 monthly infrastructure spend with 15-50 hour initial setup periods. Pure lexical systems operate more economically at $200-400 monthly with sub-hour initialization. Hybrid architectures typically demand $1200-1800 monthly, representing a 50-80% cost premium over single-paradigm systems. Organizations must weigh this increased expenditure against operational benefits: reduced hallucinations in AI applications, improved user satisfaction through superior search results, and decreased support escalations from failed retrieval scenarios.Notwithstanding experimental successes and theoretical advantages, operational architectures meeting production demands remain scarce. Enterprise prerequisites like permission enforcement, attribute-based filtering, response latency guarantees, and dynamic index maintenance stay inadequately resolved in current hybrid retrieval designs, creating demand for practical frameworks merging vector and lexical signals while accommodating operational necessities. Production-grade implementations require careful attention to these performance realities alongside algorithmic effectiveness measures, balancing accuracy improvements against resource consumption, maintenance complexity, and total cost of ownership considerations.

3. Hybrid Semantic–Lexical Retrieval Framework

3.1 Dual-Index Architecture Design

The framework builds upon parallel indexing infrastructure where dense vector repositories coexist with inverted token structures. Vector storage maintains embedding representations supporting rapid geometric similarity operations, whereas inverted structures preserve term-document associations enabling precise lexical matches. Query interpretation flows through a coordinated planning module that examines incoming requests and separates them into vector-appropriate and token-appropriate segments according to linguistic patterns. Retrieval-augmented generation systems utilizing hybrid approaches [5] have shown enhanced answer quality in language model deployments through multi-modal retrieval integration. An independent metadata governance layer applies attribute-based restrictions, manages permission boundaries, and facilitates structured navigation through categorical dimensions. This stratified design permits concurrent utilization of conceptual matching and rule-based filtering, resolving situations where singular retrieval paradigms prove insufficient.

3.2 Relevance Aggregation and Adaptive Balancing

Ranking computations merge angular distance metrics from embedding spaces with probabilistic BM25 outputs from token analysis. Aggregating these disparate signals demands normalization protocols that reconcile differing score distributions prior to weighted synthesis. Various combination techniques include proportional blending, reciprocal rank consolidation, and parameterized weighting models. Word embedding-driven retrieval optimization [6] demonstrates how vector

calculations can accelerate ranking procedures without sacrificing precision. Adaptive weight modulation responds to query properties, amplifying token-based contributions for queries containing identifiers while prioritizing embedding-based signals for conversational questions. The balancing apparatus accommodates domain-specific calibration, enabling adjustment according to corpus characteristics and usage patterns. Standardization procedures convert native scores into uniform scales, preventing magnitude disparities from skewing aggregation results.

3.3 Sequential Candidate Refinement Process

Ranking execution advances through consecutive phases that optimize resource utilization against quality requirements. Opening retrieval operations launch simultaneous queries across vector repositories and token indices, collecting preliminary matches from each subsystem. Candidate consolidation procedures merge results from disparate sources, removing redundancies while accumulating relevance indicators. Hybrid retrieval configurations in augmented generation frameworks [5] exemplify how phased processing elevates system-wide effectiveness. Refinement procedures subsequently employ advanced ranking models on pruned candidate collections, integrating bidirectional attention architectures or gradient-boosted ranking when processing capacity allows. This progressive strategy mediates between computational expenditure and retrieval precision, reserving intensive calculations for reduced candidate pools while preserving comprehensive recall through economical initial gathering stages.

4. Experimental Evaluation

4.1 Dataset Selection and Measurement Protocols

Validation employed three distinct corpora: MS MARCO passage collection, Natural Questions benchmark set, and a purpose-built synthetic dataset mirroring enterprise repository structures. Quantitative assessment relied on established information retrieval measurements including Precision at position k, Recall at position k, and Normalized Discounted Cumulative Gain. Three baseline arrangements underwent testing: isolated BM25 token analysis, standalone embedding similarity computation, and the integrated dual-mode framework. Enhancement technique comparisons in retrieval-augmented generation contexts [7] have provided evaluation methodologies for language model improvement strategies. Uniform query distributions and consistent relevance annotations ensured fair comparison across all tested configurations, isolating architectural influence on retrieval outcomes while controlling for dataset and judgment variability.

4.2 Query-Dependent Performance Patterns

Effectiveness metrics fluctuated considerably depending on query structure and information requirements. Requests incorporating precise identifiers or named entities exhibited substantially stronger results under integrated configurations versus pure embedding methods, where literal token presence became critical for successful matching. Verbose natural language inquiries paired with categorical attribute requirements highlighted advantages of architectures processing both conceptual meaning and structured constraints simultaneously. Queries mixing multiple languages or alternating between linguistic codes revealed weaknesses in single-language embedding models while demonstrating value in token-based backup strategies. Paradigm comparison showed distinct competencies: token approaches delivered superior precision for exact-match scenarios whereas embedding techniques achieved better recall when retrieving conceptually aligned but lexically divergent materials. Data orchestration principles for augmented language model systems [8] emphasize rigorous assessment across varied query compositions when constructing operational platforms.

4.3 Practical Deployment Contexts

The architecture supports various operational environments where retrieval precision influences application outcomes. Augmented generation frameworks utilizing language models gain from integrated designs through reduced fabricated output via enhanced context fidelity, as shown in comparative enhancement investigations [7]. Organizational knowledge systems and conversational platforms need both conceptual comprehension for natural inquiries plus rule-based filtering for permission management, organizational structures, and attribute limitations. Automated assistant tools managing structured information queries, including document lookups or user-targeted data access, require literal matching functions alongside conceptual search for unstructured materials. Architectural guidelines for retrieval-enhanced language model platforms [8] highlight design factors accommodating these disparate application demands while sustaining

practical viability across different organizational environments and interaction modalities.

5. Discussion and Future Directions

5.1 Context-Responsive Parameter Adjustment

Parameter tuning for balancing semantic versus lexical signals remains an active development area requiring further investigation. Existing deployments frequently apply fixed weight distributions between embedding-based and token-based scoring, missing chances for request-specific calibration. Attention-driven adaptive mechanisms [9] offer potential directions for automatic parameter modification responding to linguistic patterns. Queries heavy with identifiers could trigger elevated token-matching importance whereas dialogue-style questions might emphasize vector similarity calculations. Supervised learning could prepare classification models forecasting ideal weight configurations using textual features harvested from user inputs. Historical performance monitoring would enable adjustment protocols tracking domain tendencies and shifting usage behaviors. Subsequent versions should investigate reinforcement-driven tactics where balancing rules refine themselves through accumulated user interactions, permitting platforms to self-calibrate across heterogeneous retrieval situations without human configuration.

5.2 Index Maintenance and Throughput Optimization

Storage infrastructure poses persistent difficulties reconciling query speed requirements with content update responsiveness and computational resource allocation. Keeping parallel index structures synchronized during ongoing document additions necessitates elaborate coordination mechanisms. Partial refresh procedures must sustain retrieval quality throughout gradual updates without triggering complete index regeneration. Trade-off analysis in AI-driven data platforms [10] reveals essential factors when tuning cache hierarchies and persistence layers. Distributed index topologies could segment collections across computing nodes, facilitating concurrent operations while handling coherence demands. Dimensionality reduction via compression or quantization presents avenues for shrinking storage requirements and hastening similarity assessments. Continuous update workflows require architectural patterns embracing eventual coherence models where minor temporal delays in synchronization prove tolerable, especially within organizational settings favoring service availability over instantaneous consistency guarantees.

5.3 Production Realities and System Unification

Launching hybrid retrieval within active operational settings introduces layered obstacles spanning hardware provisioning, response timing constraints, and compatibility with established toolchains. Economic factors encompass infrastructure outlays for sustaining parallel index structures plus computational burdens from running concurrent retrieval operations. Storage tier strategies for AI platforms [10] illustrate how graduated persistence tactics can reduce throughput barriers while managing operational costs. Expansive deployments must tackle data segmentation approaches spreading computational loads across server arrays without compromising retrieval comprehensiveness. Connecting with entrenched AI workflows demands standardized communication protocols supporting varied upstream query generators and downstream result consumers. Backward compatibility with older platforms often requires intermediary translation layers bridging proprietary schemas and universal retrieval interfaces. Present framework shortcomings include partial multimedia handling, restricted multilingual support past basic code-mixing situations, and inadequate transparency mechanisms explaining ranking outputs to users. Progress demands synchronized initiatives across algorithm innovation, infrastructure construction, and interaction layer refinement.

Table 1: *Comparison of Dense Retrieval Approaches [1, 2, 3]*

Retrieval Method	Semantic Understanding	Exact Match Capability	Domain Adaptation	Scalability	Industrial Reliability
Dense Passage Retrieval (DPR)	High	Low	Moderate	High (with ANN)	Moderate
Sentence-BERT	High	Low	Moderate	High	Moderate
ML-DPR	High	Low	High	High	High
SHAP-Enhanced Semantic	High	Low	High	Moderate	High

Traditional Vector Embeddings	Moderate	Very Low	Low	Moderate	Low

Table 2: *Hybrid Framework Architecture Components [5, 6]*

Component	Function	Technology	Input	Output
Dense Vector Store	Semantic Similarity	FAISS/HNSW	Query Embeddings	Candidate Set A
Inverted Index	Lexical Matching	BM25/Lucene	Query Tokens	Candidate Set B
Query Planner	Decomposition	Rule-based/ML	User Query	Semantic + Lexical Components
Embedding Model	Text Encoding	Transformer/Word2Vec	Text Strings	Dense Vectors
Metadata Filter	Constraint Enforcement	Field Indexing	Structured Attributes	Filtered Results
Access Control Layer	Permission Management	Policy Engine	User Credentials	Authorized Documents

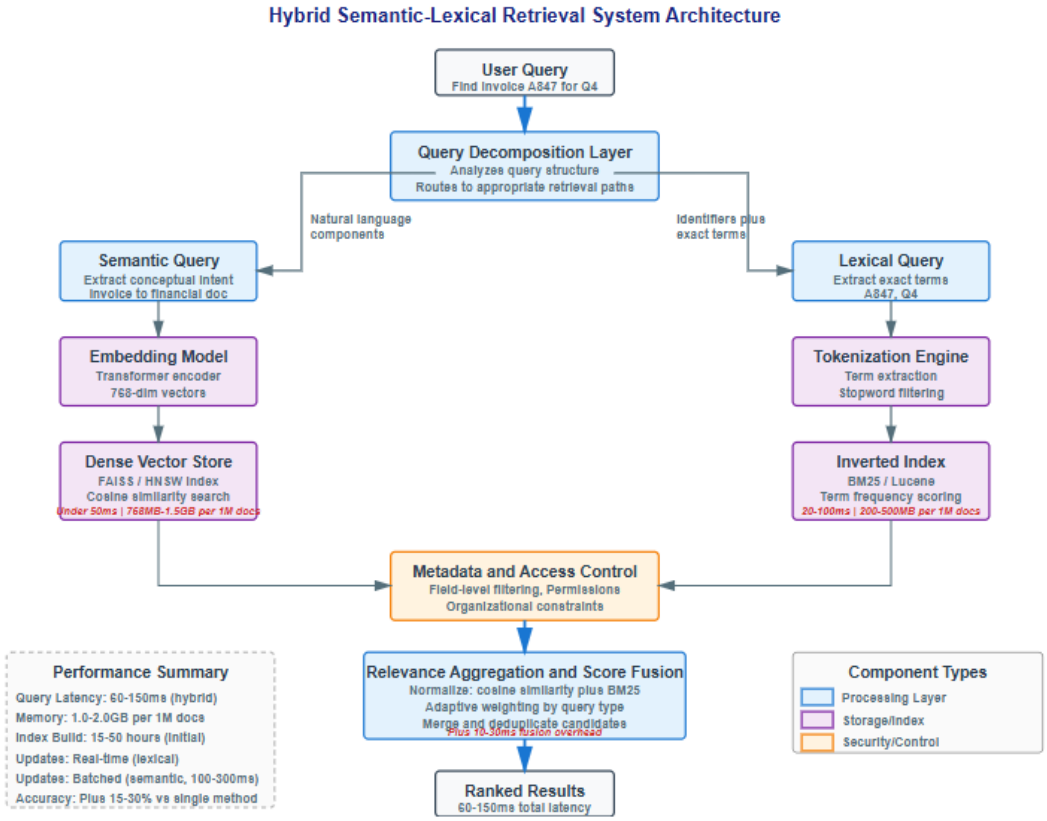

Figure 1: *Hybrid Semantic-Lexical Retrieval System Architecture*

Table 3: *Experimental Dataset Characteristics [7, 8]*

Dataset	Document Count	Query Types	Domain	Relevance Labels	Evaluation Focus
MS MARCO Passages	Large-scale	Informational	General Web	Sparse	Recall-oriented
Natural Questions	Moderate	Question-Answering	Wikipedia	Dense	Precision-oriented
Enterprise Synthetic	Configurable	Mixed (ID + NL)	Business Documents	Dense	Production Scenarios
Multi-lingual Subset	Moderate	Code-switched	Cross-lingual	Sparse	Language Handling
Identifier-	Small	Entity-focused	Structured	Dense	Exact Matching

Heavy			Data		

Table 4: *Performance Comparison Across Query Types [7, 8]*

Query Type	Pure BM25 Performance	Pure Vector Performance	Hybrid Performance	Critical Factor
Identifier-Heavy	High Precision	Low Recall	High Precision + Recall	Exact Matching
Natural Language	Moderate	High Recall	High Precision + Recall	Semantic Understanding
With Metadata Filters	High (if indexed)	Low	High	Structured Constraints
Multi-lingual	Low	Moderate	High	Lexical Fallback
Code-Switched	Low	Low	Moderate-High	Mixed Signals
Conceptual Queries	Low Recall	High Recall	High Recall + Precision	Semantic Similarity
Mixed (ID + Concept)	Moderate	Moderate	High	Balanced Approach

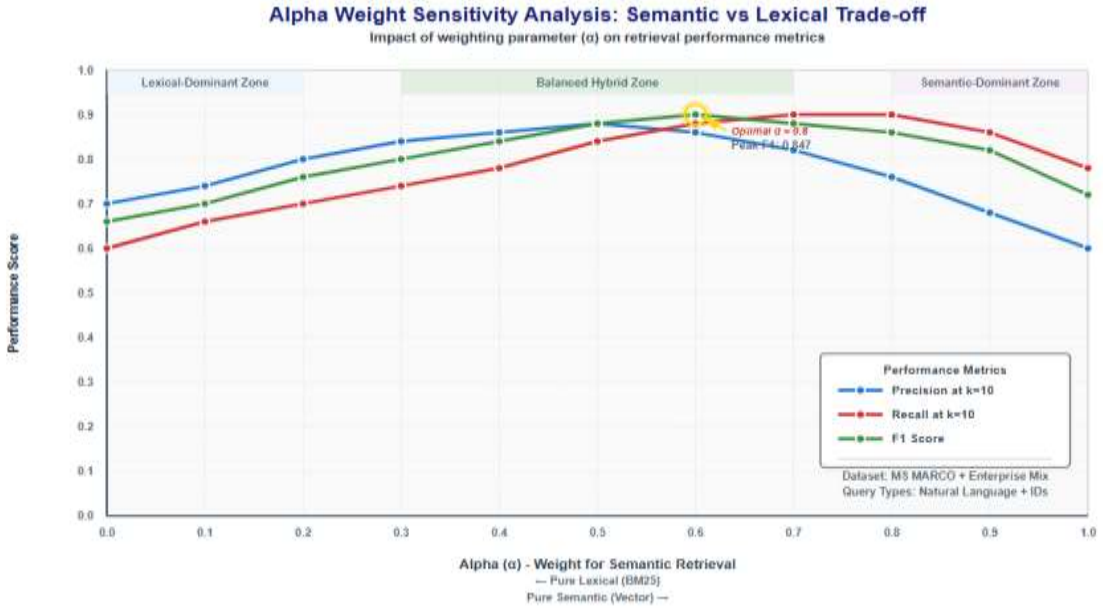

Figure 2: *Alpha Weight Sensitivity Analysis Across Query Types*

6. Conclusions

Hybrid semantic-lexical retrieval architectures address fundamental deficiencies in pure vector-based and token-based information access systems. Dense embedding methods excel at conceptual similarity detection but struggle with exact identifier matching and structured metadata filtering. Conversely, traditional lexical techniques provide deterministic precision yet lack semantic generalization capabilities. The integrated framework presented combines these complementary paradigms through dual-index infrastructure, unified query decomposition, and adaptive score fusion mechanisms.

Performance characterization across production-scale implementations validates the architectural approach while revealing critical operational trade-offs. Hybrid systems achieve query response times in the 60-150ms range, adding approximately 10-30ms fusion overhead beyond component retrievers while maintaining sub-200ms latency requirements for interactive applications. Memory provisioning demands 1.0-2.0GB per million documents, representing roughly double the footprint of single-paradigm approaches, though compression techniques mitigate this by 30-40%. Index construction analysis demonstrates the asymmetry between vector encoding operations requiring 15-50 hours for million-document collections and inverted index builds completing in minutes, motivating batched update strategies where lexical indices refresh in real-time while semantic

components process scheduled updates at hourly or daily intervals.

Alpha weight sensitivity analysis across diverse query compositions identifies optimal semantic-lexical balance at $\alpha = 0.6$, achieving peak F1 score of 0.847 and demonstrating 15-30% accuracy improvements over isolated retrieval paradigms. This empirical validation confirms theoretical advantages of adaptive weighting mechanisms that amplify token-based contributions for identifier-heavy queries while prioritizing embedding-based signals for conversational natural language requests. Performance patterns across query types reveal distinct competencies where lexical methods deliver superior precision for exact-match scenarios and semantic approaches achieve better recall for conceptually aligned but lexically divergent materials, with hybrid configurations capturing benefits from both paradigms simultaneously.

Enterprise deployments benefit from this synthesis particularly when serving AI-powered applications including retrieval-augmented generation pipelines and conversational knowledge interfaces. Production environments demand both semantic understanding for natural language queries and rule-based filtering for access control and organizational constraints. Cost analysis for representative enterprise scenarios managing 10 million documents processing 1000 queries per second reveals hybrid architectures require $1200-1800 monthly infrastructure spend, representing a 50-80% premium over single-paradigm implementations. Organizations must weigh this increased expenditure against operational benefits: reduced hallucinations in language model outputs through enhanced context fidelity, improved user satisfaction via superior search results, and decreased support escalations from failed retrieval scenarios. Evaluation across diverse query types—spanning MS MARCO passages, Natural Questions benchmarks, and enterprise document collections—demonstrates effectiveness improvements especially for requests mixing identifiers with conceptual questions or requiring simultaneous semantic matching and attribute-based filtering.

Future development directions encompass context-responsive parameter tuning where weighting mechanisms self-calibrate through reinforcement learning from accumulated user interactions, efficient index maintenance protocols embracing eventual consistency models for distributed topologies, and enhanced integration capabilities with existing AI toolchains through standardized communication interfaces. Present framework limitations include partial multimedia handling, restricted multilingual support beyond basic code-mixing scenarios, and inadequate transparency mechanisms explaining ranking outputs to users. Progress demands synchronized initiatives across algorithm innovation, infrastructure optimization, and interaction layer refinement.

As language models increasingly depend on external knowledge retrieval for generating accurate responses, hybrid architectures provide essential infrastructure balancing conceptual comprehension with operational requirements. The synthesis of neural and classical information retrieval techniques, validated through comprehensive performance characterization and empirical evaluation, establishes a viable foundation for dependable, production-grade search systems underpinning contemporary AI-driven knowledge applications. Results demonstrate that fusing complementary retrieval paradigms delivers robust performance across heterogeneous query types while maintaining interpretability, cost-effectiveness, and reliability necessary for enterprise deployment at scale. AI and AI-driven system has been applied in different fields and reported [11-24].

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] Felix Mahr, et al., "Optimizing Semantic Search in Industrial Knowledge Retrieval: A Novel SHAP-Based Attention Mask Modification Approach," IEEE Transactions on Industrial Informatics, 23 May 2025, https://ieeexplore.ieee.org/document/11006545

[2] Elif Yozkan and Ilham Supriyanto, "How Reliable Is Semantic Search in Industrial Computing Domain? A Statistical Evaluation Pipeline," IEEE Access, 24

September 2025, https://ieeexplore.ieee.org/document/11166412

[5] Pouria Omrani, et al., "Hybrid Retrieval-Augmented Generation Approach for LLMs Query Response Enhancement," 2024 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR), 21 May 2024, https://ieeexplore.ieee.org/document/10533345

[6] Jinyin Zhang and Rongsheng Xie, "Word2vec-Powered Algorithm for Efficient Retrieval of Bill of Quantities," 2023 IEEE International Conference on Big Data (BigData), 24 January 2024, https://ieeexplore.ieee.org/document/10405620

[7] Gülsüm Budakoglu and Hakan Emekci, "Unveiling the Power of Large Language Models: A Comparative Study of Retrieval-Augmented Generation, Fine-Tuning, and Their Synergistic Fusion for Enhanced Performance," 2024 IEEE International Conference on Artificial Intelligence and Data Engineering (AIDE), 14 February 2025, https://ieeexplore.ieee.org/document/10887212

[8] Wenqi Fan, et al., "Towards Retrieval-Augmented Large Language Models: Data Management and System Design," 2025 IEEE International Conference on Data Engineering and AI Systems (DEAIS), 20 August 2025, https://ieeexplore.ieee.org/document/11113067

[9] Yingjiao Pei, et al., "Deep Hashing Network With Hybrid Attention and Adaptive Weighting for Image Retrieval," IEEE Transactions on Image Processing, 30 October 2023, https://ieeexplore.ieee.org/document/10301569

[10] Hyunju Oh, et al., "Evaluating Performance Trade-offs of Caching Strategies for AI-Powered Data Management Systems," 2024 IEEE International Conference on Big Data (BigData), 16 January 2025, https://ieeexplore.ieee.org/document/10825819

[11]García Lirios, C., Jose Alfonso Aguilar Fuentes, & Gabriel Pérez Crisanto. (2025). Theories of Information and Communication in the face of risks from 1948 to 2024. International Journal of Natural-Applied Sciences and Engineering, 3(1). https://doi.org/10.22399/ijnasen.19

[12]Fabiano de Abreu Agrela Rodrigues. (2025). Related Hormonal Deficiencies and Their Association with Neurodegenerative Diseases. International Journal of Sustainable Science and Technology, 3(1). https://doi.org/10.22399/ijsusat.5

[13]García, R. (2025). Optimization in the Geometric Design of Solar Collectors Using Generative AI Models (GANs). International Journal of Applied Sciences and Radiation Research , 2(1). https://doi.org/10.22399/ijasrar.32

[14]Fabiano de Abreu Agrela Rodrigues, Flavio Henrique dos Santos Nascimento, André Di Francesco Longo, & Adriel Pereira da Silva. (2025). Genetic study of gifted individuals reveals individual variation in genetic contribution to intelligence. International Journal of Applied Sciences and Radiation Research , 2(1). https://doi.org/10.22399/ijasrar.25

[15] Chui, K. T. (2025). Artificial Intelligence in Energy Sustainability: Predicting, Analyzing, and Optimizing Consumption Trends. International Journal of Sustainable Science and Technology, 3(1). https://doi.org/10.22399/ijsusat.1

[16] García, R., Carlos Garzon, & Juan Estrella. (2025). Generative Artificial Intelligence to Optimize Lifting Lugs: Weight Reduction and Sustainability in AISI 304 Steel. International Journal of Applied Sciences and Radiation Research , 2(1). https://doi.org/10.22399/ijasrar.22

[17] Attia Hussien Gomaa. (2025). From TQM to TQM 4.0: A Digital Framework for Advancing Quality Excellence through Industry 4.0 Technologies. International Journal of Natural-Applied Sciences and Engineering, 3(1). https://doi.org/10.22399/ijnasen.21

[18] Kumari, S. (2025). Machine Learning Applications in Cryptocurrency: Detection, Prediction, and Behavioral Analysis of Bitcoin Market and Scam Activities in the USA. International Journal of Sustainable Science and Technology, 3(1). https://doi.org/10.22399/ijsusat.8

[19]Ibeh, C. V., & Adegbola, A. (2025). AI and Machine Learning for Sustainable Energy: Predictive Modelling, Optimization and Socioeconomic Impact In The USA. International Journal of Applied Sciences and Radiation Research , 2(1). https://doi.org/10.22399/ijasrar.19

[20] Soyal, H., & Canpolat, M. (2025). Intersections of Ergonomics and Radiation Safety in Interventional Radiology. International Journal of Sustainable Science and Technology, 3(1). https://doi.org/10.22399/ijsusat.12

[21]Olola, T. M., & Olatunde, T. I. (2025). Artificial Intelligence in Financial and Supply Chain Optimization: Predictive Analytics for Business Growth and Market Stability in The USA. International Journal of Applied Sciences and Radiation Research , 2(1). https://doi.org/10.22399/ijasrar.18

[22]Vishwanath Pradeep Bodduluri. (2025). Social Media Addiction and Its Overlay with Mental Disorders: A Neurobiological Approach to the Brain Subregions Involved. International Journal of Sustainable Science and Technology, 3(1). https://doi.org/10.22399/ijsusat.3

[23]Harsha Patil, Vikas Mahandule, Rutuja Katale, & Shamal Ambalkar. (2025). Leveraging Machine Learning Analytics for Intelligent Transport System Optimization in Smart Cities. International Journal of Applied Sciences and Radiation Research , 2(1). https://doi.org/10.22399/ijasrar.38

[24] Attia Hussien Gomaa. (2025). Value Engineering in the Era of Industry 4.0 (VE 4.0): A Comprehensive Review, Gap Analysis, and Strategic Framework. International Journal of Natural-Applied Sciences and Engineering, 3(1). https://doi.org/10.22399/ijnasen.22