EFRA
EXTREME FOOD RISK ANALYTICS

Best Practices for Ontology-Aware Retrieval in LLM-Based Systems

December 2025

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Commission-EU. Neither the European Union nor the granting authority can be held responsible for them

1. Introduction

Large Language Models (LLMs) have significantly changed the way semantic annotation, entity linking, and knowledge extraction systems can be designed. Instead of relying exclusively on fully supervised classifiers or rule-based pipelines, modern architectures increasingly adopt *Retrieval-Augmented Generation* (RAG) patterns, where structured knowledge bases are queried at runtime and the results are interpreted by an LLM. This paradigm was originally formalised in the context of open-domain question answering (Lewis et al., 2020) and has since been extended to a wide range of knowledge-intensive NLP tasks.

Within the Horizon Europe **EFRA** project, this paradigm is applied to food safety incident analysis, where large, heterogeneous ontologies (e.g. AGROVOC[1], FoodOn[2], ChEBI[3]) are used as authoritative conceptual backbones. The scale and complexity of these resources—ranging from tens of thousands to hundreds of thousands of concepts—make naïve prompting approaches infeasible. Prior work has shown that large label spaces severely degrade both supervised and prompt-based classification performance (Chen et al., 2021).

Instead, search indexes over ontologies become a *critical infrastructure component* enabling efficient candidate retrieval and high-quality entity linking. Recent studies demonstrate that combining LLM reasoning with retrieval over structured knowledge substantially improves accuracy, scalability, and interpretability in annotation tasks (Lewis et al., 2020; Karpukhin et al., 2020).

This document provides **best practices for designing, building, and maintaining search indexes over ontologies** to support LLM-agent-based RAG systems for semantic annotation. While grounded in the EFRA use case, the principles are intended to be reusable across domains where structured knowledge and LLMs are combined.

The document targets technical architects, researchers, and developers working on AI-driven knowledge extraction pipelines in EU research and innovation projects.

[1] https://agrovoc.fao.org/

[2] https://foodon.org/

[3] https://www.ebi.ac.uk/chebi/

2. Design Principles

Before addressing implementation details, it is essential to define a set of guiding principles that should shape any ontology indexing strategy for LLM-based systems. These principles are informed both by EFRA experimentation and by recent advances in LLM-based text analysis and retrieval-augmented systems.

2.1 Retrieval First, Reasoning Second

In a RAG-like entity linking architecture, the search index is **not** expected to make the final semantic decision. Its role is to:

- Maximize *recall* of potentially relevant concepts
- Provide *rich, interpretable context* to the LLM
- Reduce the effective label space to a manageable candidate set

The LLM is responsible for disambiguation and final selection. Empirical evidence shows that LLMs perform best when reasoning over a small, well-curated candidate set rather than over entire taxonomies or ontologies (Wei et al., 2022). Therefore, indexing strategies should favor *inclusive retrieval* over overly strict precision.

2.2 Ontologies as Knowledge Objects, Not Just Labels

Ontology concepts should not be indexed as simple strings (labels only). Instead, each concept must be treated as a **knowledge object** with multiple semantic facets:

- Lexical information (labels, synonyms, multilingual terms)
- Structural information (hierarchy, parent/child relations)
- Descriptive context (definitions, scope notes, inclusions/exclusions)
- Domain-specific metadata

This view aligns with Linked Open Data and SKOS best practices and has proven particularly important when LLMs are used for semantic interpretation rather than strict classification (W3C, 2009; Wei et al., 2022).

2.3 Hybrid Retrieval is Mandatory

Pure semantic (vector-only) or pure lexical (keyword-only) retrieval is insufficient for large, heterogeneous ontologies. Best practice is to adopt **hybrid retrieval**, combining:

- Dense vector search (semantic similarity)
- Sparse keyword search (exact and fuzzy matching)

Hybrid approaches have consistently outperformed single-mode retrieval in large label-space settings and are now considered state of the art for knowledge-intensive NLP pipelines (Liu et al.., 2023; Karpukhin et al., 2020).

3. Ontology Preparation and Normalization

Effective indexing starts *before* any search engine is configured. Ontology preparation is a decisive step.

3.1 Concept Canonicalization

Each ontology concept should be transformed into a canonical internal representation. At minimum, the following fields are recommended:

- **Concept ID**: Stable, globally unique identifier (URI preserved)
- **Preferred Label**: Official primary label
- **Alternative Labels**: Synonyms, abbreviations, lexical variants
- **Definition / Description**: Human-readable explanatory text
- **Ontology Source**: AGROVOC, FoodOn, ChEBI, GS1 GPC, etc.
- **Concept Type / Root Category**: Product, Hazard, Substance, Process, etc.

This canonicalization layer decouples downstream indexing from ontology-specific serialization formats (RDF, OWL, SKOS).

3.2 Textual Enrichment

Ontology concepts often suffer from sparse textual descriptions, which can limit their retrievability in search and retrieval systems. To address this, several strategies can be employed:

- **Aggregate descriptive text:** Concatenate concept labels, synonyms, and formal definitions into a single searchable text field to maximize coverage of possible query terms.

- **Normalize text:** Standardize capitalization, punctuation, and Unicode representations to reduce retrieval errors caused by surface form variations.

- **Expand abbreviations:** Where domain knowledge allows, expand common abbreviations or acronyms to their full forms (e.g., “PCB” → “polychlorinated biphenyls”) to improve discoverability.

For resources with a richer semantic structure, such as GS1 GPC[4] where inclusion and exclusion notes are especially valuable for capturing contextual boundaries not reflected in labels or definitions, best practice is **to always index these notes,** making them fully searchable alongside other descriptive text to enhance both precision and recall in concept retrieval.

3.3 Language Handling

Preserve language tags for labels where available by either:

- Indexing each language separately, or
- Creating multilingual embeddings using language-agnostic models Mixing

languages without explicit strategy significantly degrades retrieval quality.

4. Index Architecture

4.1 Logical Index Separation

Best practice is to logically separate indexes along **semantic responsibility lines**, for example:

- Product concepts
- Hazard concepts
- Chemical substances
- Processes and treatments

This enables:

- Targeted querying by the LLM agent
- Reduced noise in candidate retrieval
- Domain-specific tuning of retrieval parameters

Physical separation (multiple indexes) or logical filtering (single index with strong faceting) are both acceptable, depending on infrastructure constraints.

[4] https://gpc-browser.gs1.org/

4.2 Field-Level Indexing Strategy

Each concept should be indexed using multiple fields, with different retrieval roles:

Field	Purpose
preferred_label	High-precision lexical matching
alternative_labels	Recall expansion
description	Semantic grounding for embeddings
ontology_source	Filtering and traceability
hierarchy_context	Disambiguation support

The hierarchy context may include parent labels or top-level categories concatenated as text.

4.3 Vector Embeddings - Model Selection

Embedding models should:

- Be domain-tolerant (food, chemistry, products)
- Support multilingual input if required
- Be stable over time (to avoid frequent re-indexing)

Consistency across all indexed concepts is more important than marginal gains from frequent model changes.

A recommended best practice is to embed a **composite textual representation**, for example:
Preferred label + synonyms + short definition + parent category

This improves semantic clustering and reduces false positives.

5. Hybrid Retrieval Configuration

Keyword-based search should be capable of identifying exact matches as well as near matches using fuzzy techniques, such as edit-distance calculations, and should support phrase queries. To improve relevance, ranking strategies can be applied that prioritize certain fields, for example: *Preferred label > alternative labels > description*

Vector-based retrieval should leverage similarity metrics such as cosine similarity (or equivalent) and return a configurable number of top candidates, typically in the range of 20–50. Extremely small top-K values should be avoided, as they increase the risk of excluding correct concepts.

Hybrid retrieval, combining keyword and vector approaches, requires a fusion strategy to integrate scores effectively. This can be achieved through weighted score

combinations, reciprocal rank fusion, or a two-stage approach where a keyword-based filter generates a candidate set that is subsequently re-ranked using vector similarity.

The objective of these retrieval strategies is not to produce a perfect ranking but to generate **high-quality candidate sets** that provide reliable input for downstream LLM reasoning and decision-making.

6. Index Outputs for LLM Consumption

6.1 LLM-Friendly Result Schema

Search results should be returned in a structured, compact format, for example:

- Concept ID
- Label
- Short description
- Ontology source
- Optional hierarchy path

Avoid returning raw search engine metadata or excessively long texts.

6.2 Context Budget Awareness

LLMs have finite context windows. Best practices include:

- Limiting candidate count per query
- Truncating descriptions to the most informative segments
- Avoiding redundant synonyms

The index should support configurable verbosity levels depending on the task.

7. Integration with LLM Agents

7.1 Search as a Tool

Within EFRA-like architectures, search functions should be exposed to the LLM agent as explicit tools with:

- Clear input contracts (text fragment, concept type)
- Deterministic outputs
- Traceable provenance

This enables transparent reasoning chains and reproducibility.

7.2 Iterative Retrieval

Agents should be allowed to:

- Reformulate queries
- Query different semantic indexes
- Combine results across ontologies

Index design must therefore support low-latency, repeatable queries.

8. Evaluation and Maintenance

8.1 Retrieval-Level Evaluation

Evaluation should not focus only on final entity linking accuracy. Retrieval-level metrics are essential:

- Recall@K for gold entities
- Candidate set diversity
- Cross-ontology coverage

8.2 Versioning and Provenance

Ontologies evolve. Best practice includes:

- Versioned indexes
- Explicit ontology version metadata
- Reproducible indexing pipelines

This is critical for scientific transparency in EU research projects.

9. Common Pitfalls and Anti-Patterns

- Indexing labels only, without definitions
- Using a single monolithic index without semantic filtering
- Over-optimizing ranking instead of recall
- Treating LLMs as search engines

Avoiding these pitfalls significantly improves system robustness.

10. Conclusions and Recommendations

Search indexes over ontologies are **first-class citizens** in LLM-based semantic annotation architectures. In EFRA-like RAG systems, they act as the bridge between symbolic knowledge and neural reasoning.

Key recommendations:

1. Treat ontology concepts as rich knowledge objects rather than flat labels
2. Use hybrid retrieval by design, not as an afterthought
3. Optimize for recall and interpretability, not perfect ranking
4. Design indexes for LLM consumption, not human browsing
5. Ensure reproducibility, versioning, and provenance tracking

By following these best practices, EU research projects can build scalable, transparent, and future-proof knowledge bases that fully leverage the strengths of LLM agents while remaining grounded in authoritative semantic resources.

References

Chen, H., Ma, Q., Lin, Z., & Yan, J. (2021).

Hierarchy-aware label semantics matching network for hierarchical text classification. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL-IJCNLP).*

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020).

Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.

Advances in Neural Information Processing Systems (NeurIPS).

Wei, J., Bosma, M., Zhao, V. Y., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., & Le, Q. V. (2022).

Finetuned Language Models Are Zero-Shot Learners.

Transactions on Machine Learning Research.

Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W. (2020).

Dense Passage Retrieval for Open-Domain Question Answering.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP).

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023).

Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv preprint arXiv:2312.10997.*

Miles, A., & Bechhofer, S. (2009*).* SKOS Simple Knowledge Organization System Reference.

W3C Recommendation.