Annals of Library and Information Studies
Vol. 72, December 2025, pp. 446-469
DOI: 10.56042/alis.v72i4.25678

Applying S.R. Ranganathan's Classification Theory to Investigate the Epistemology of Knowledge Organization in Large Language Models (LLMs)

Dr. R. J. Kalpana[a] and Yuvaraj Athur Raghuvir[b]
[a]Founder, Pen and Ink Solutions Email : kalpanarj@gmail.com
[b]Dyut Inc, E mail: Yuva.athur@dyut.dev

Received: 16 May 2025; Accepted: 13 November 2025

It has been explored that how S.R. Ranganathan's faceted classification (Colon Classification) and epistemological ideas can inform the design and validation of Large Language Models (LLMs) . Ranganathan's five fundamental categories - Personality, Matter, Energy, Space, Time (PMEST)—and his three planes of work (Idea, Verbal, Notational) offer a framework for organising knowledge that can anchor and verify AI outputs. A conceptual model in which every LLM-generated claim is mapped onto a structured faceted scheme or knowledge graph (KG) via a Faceted Classification Module (FCM) and a Classification Validator is proposed. This validator tags each claim with PMEST facets, checks semantic and epistemic consistency using modern NLP techniques (detailed in Appendix A), and rates certainty using a seven-level epistemic hierarchy (detailed in Appendix B). Using the query "Uses of turmeric in Indian medicine" as a running example, it is demonstrated how Ranganathan's planes of work apply to LLM interactions and illustrate each pipeline step with technical specifications. This approach has been compared with the existing KG-augmented RAG methods, particularly Microsoft's GraphRAG. It has been further discussed how classification-based systems complement graph-based retrieval for improved fact-checking and cultural context preservation. Recent empirical findings show that advanced LLMs suffer high hallucination rates (GPT-4: ~29% hallucinations, ~13% precision on scientific reference tasks) and that knowledge graph-based approaches yield substantial improvements in complex question-answering performance. Overall, it is concluded that LLM knowledge organisation is a bibliographic process grounded in library epistemology, aiming to detect hallucinations, enhance explainability, and integrate diverse knowledge systems—including Indian Knowledge Systems—systematically.

Keywords: S.R. Ranganathan, Faceted Classification, Knowledge Organisation Systems (KOS), Large Language Models (LLMs), Epistemology, Knowledge Graphs, Taxonomies, LLM Validation, Retrieval-Augmented Generation (RAG), GraphRAG, Culturally Rooted AI, Explainable AI (XAI), Indian Knowledge Systems (IKS), Ontology Integration

Introduction

Classification has long been central to organising human knowledge. Ranganathan's Colon Classification revolutionised knowledge organisation by introducing an analytico-synthetic (faceted) system, treating any subject as a combination of its facets : Personality, Matter, Energy, Space, and Time (PMEST) which he termed as fundamental categories. He viewed a library as an "ever-growing organism" that can dynamically accommodate new books on different subjects without breaking existing order [1]. In the age of AI, scholars increasingly consult large language models (LLMs) for answers to complex queries, but these models often hallucinate or give inconsistent information, especially in specialised or cross-cultural domains[2,3] . Unlike books on shelves arranged by subjects according to call numbers, LLMs encode knowledge in distributed vector spaces with no explicit taxonomy. This gap raises a critical question: How can users verify the reliability of LLM responses?

This article proposes that just as a library user relies on a classification system to find trustworthy information, an LLM user needs a comparable knowledge organisation framework to assess AI outputs. Ranganathan's work provides a rich analogy: his faceted breakdown of any subject into PMEST categories can be used to systematically tag LLM claims, and his "planes of work" (Idea, Verbal, Notational) can model the flow from user intent to query to response [4] (Dousa, 2019). Our key insight is to treat an LLM's answer like a library document that must be placed in context. For example, consider the query "What are the uses of turmeric in Indian medicine?"—the user's underlying idea is the medicinal uses of turmeric (Idea plane). The actual prompt phrasing is the Verbal plane. The Notational plane would assign classification facets (e.g., Main Class: Medicine, System: Ayurveda, Personality: Turmeric; Matter: Anti-inflammatory; Energy: Healing process; Space: India;), which anchor the answer in a structured scheme. By mapping each claim of the answer onto such facets or a knowledge graph, a validator can check factuality and coherence.

In this paper, a conceptual framework that applies Ranganathan's facet analysis, planes of work, and fundamental categories to LLM design and use has been developed, while acknowledging extensions proposed by the Classification Research Group (CRG) and contemporary facet theorists [5-7]. Some recent case studies in the subject have been included. For instance, empirical evaluations of LLM reliability reveal serious shortcomings: Chelli et al. (2024) [2] found out that GPT-4's precision was only 13.4% when generating citations for systematic reviews, with a hallucination rate of 28.6%. Similarly, Mahaut et al. [3] (2024) demonstrate that confidence estimates for LLM facts are often inconsistent and unstable under paraphrase. These findings underscore the need for external verification layers.

The article introduces a seven-level epistemic hierarchy (from unsupported opinion to proven fact) for grading LLM assertions, grounded in evidence-based practice frameworks such as GRADE and the Oxford Centre for Evidence-Based Medicine levels [8-9]. Each level is explicitly defined with implementation guidelines and concrete examples in Appendix B. A practical model is proposed with two components: (1) a Faceted Classification Module (FCM)—an external knowledge organisation system that encodes subjects in PMEST terms—and (2) an LLM Validator/Classifier that parses each LLM output using modern Natural Language Processing (NLP) techniques. The specific NLP/ML approaches for automatic facet identification and tagging are detailed in Appendix A, including Named Entity Recognition models, relation extraction tools, ontology alignment methods, and graph query mechanisms.

The proposed model is compared with the recent efforts that integrate knowledge graphs with LLMs. Microsoft's GraphRAG system builds a knowledge graph from documents and uses it to augment LLM queries [10]. It is explained how GraphRAG's retrieval-augmented generation (RAG) approach differs from the classification-based validation and how the two can complement each other. Throughout, the running example of turmeric in Indian medicine is used to explain the ground abstract concepts in concrete application.

Literature Review

Ranganathan's Faceted Classification and Its Extensions

S.R. Ranganathan (1892-1972) is celebrated for his dynamic theory of classification and Colon Classification that can be extended for knowledge organisation. He introduced five fundamental categories - Personality (P), Matter (M), Energy (E), Space (S), and Time (T)—as universal dimensions of any subject (Ranganathan, 1967). For example, "dental surgery" is classified by combining Medicine as the Main Subject, Teeth (P), Disease, and Surgery (E) into the notation L 214:4:7. Unlike enumerative schemes (e.g., Dewey Decimal CLassification) that list subjects statically, Ranganathan's analytico-synthetic method allows on-the-fly composition of subjects using appropriate connecting symbols between facets. This makes Colon Classification highly flexible: new combinations can be added as knowledge grows, without reorganising the system. Ranganathan argued that a thorough faceted scheme mirrors human thought and rationality, enabling precise pinpointing of any topic.

The Classification Research Group (CRG), active in the UK from the 1950s onwards, built upon and extended Ranganathan's work (Classification Research Group, 1978). While recognszing PMEST as foundational, the CRG developed domain-specific facet structures for areas such as music, architecture, and engineering, acknowledging that different disciplines may require additional or alternative facets [5]. Spiteri (1998) [6] proposed a simplified model for facet analysis that emphasises empirical domain analysis over purely rationalistic approaches. Tennis (2008) [7] explored how form and function interact in classification, noting that facet structures must adapt to the domains they describe. Hjørland (2014) [11] questioned whether facet analysis is purely rationalistic or must be grounded in empirical observation of domain structures, highlighting ongoing epistemological debates in classification theory.

For the purpose of this article, PMEST has been adopted as a foundational framework while explicitly acknowledging its limitations and the need for domain-specific extensions. As Hjørland (2014) [11] notes, classification schemes must balance rationalistic structure with empirical responsiveness.

Our Faceted Classification Module (FCM) is designed to accommodate both Ranganathan's universal facets and the CRG's domain-specific elaborations. This flexibility allows the system to handle diverse subject areas—from Ayurvedic Medicine to Quantum Physics—without forcing all knowledge into a procrustean bed.

Philosophically, Ranganathan saw classification as an epistemological act—a reflection of how knowledge is understood and structured. He distinguished intellection (logical processing of sensory data) from intuition (direct insight into reality), and regarded facets as expressions of rational order in knowledge organization [4]. This work builds on this view: facet tagging and classification have been treated as a means of bringing LLM outputs into a humanly interpretable knowledge structure.

LLM Knowledge and Reliability

Modern LLMs (e.g., GPT-4, PaLM, Claude) store knowledge in high-dimensional vectors and generate text by pattern completion. While they excel at fluent language and broad knowledge recall, numerous studies show they also hallucinate facts, omit details, or give conflicting answers. For instance, an LLM may confidently assert an incorrect chemical property or invent a citation. This unpredictability is especially problematic in specialised domains (medicine, law, local knowledge) and when subtle nuances matter. Chelli et al. [2](2024) found that GPT-4 could retrieve less than 14% of correct references in a systematic literature review task, with over 28% of its outputs being entirely fabricated. Mahaut et al. [3] (2024) show that LLM confidence scores for factual claims are unstable under paraphrasing, suggesting shallow understanding.

Attempts to improve LLM factuality include better training data, fine-tuning, and retrieval-augmented generation (RAG) methods that supply external documents. RAG can ground an answer by quoting real sources, but vector retrieval alone is not foolproof—semantic search may miss relevant facts or struggle with multi-step reasoning [12]. Some authors have called for structural methods to organise LLM knowledge. For example, Zheng et al.[13] (2024) argue that LLMs behave like unstructured knowledge bases and suggest combining them with symbolic graphs. Pan et al. [14] (2024) propose a "unified" LLM+KG paradigm. In industry, tools like Microsoft's GraphRAG extract knowledge graphs from text and use them to augment queries [10] .

These efforts resonate with Ranganathan's insight: explicit knowledge structures can make AI outputs more reliable. However, little work has explicitly applied library classification theory to LLMs. The gap filled here, is to systematically map LLM outputs into a faceted classification scheme, treating each claim as a mini-"document" to be indexed and validated. This bibliographic perspective draws on both facets (as semantic dimensions) and planes of work (as stages of knowledge formation) to analyse LLM processes. It also introduces a hierarchy of evidence (from opinion to proven fact) not commonly addressed in LLM research. By bridging Ranganathan's classification epistemology with modern AI methods, it is aimed to improve LLM reliability and inclusivity, especially for culturally rooted knowledge.

Theoretical Framework

The framework here adapts Ranganathan's key concepts—facets, planes of work, classification as language, and facet growth—to the context of LLM interactions, while incorporating insights from CRG and contemporary classification theory.

Facets as Semantic Dimensions

At the heart of Ranganathan's method are the PMEST facets, which he regarded as universally present in any subject. Facets act like coordinates in a multi-dimensional "subject space," allowing precise specification of a topic [1] . It is proposed to view each claim of an LLM answer through these PMEST lenses. For any statement generated by the LLM, one can ask: What is the personality (primary entity/concept) here? What is the matter (attribute or content)? Is there an energy (action/process) described? What is the space (location/context)? What is the time (temporal context)? By explicitly assigning facet labels, an answer is decomposed into structured elements.

While we ground our framework in PMEST, we recognise that complex domains may require additional facets or hybrid schemes, as the CRG demonstrated for specialised fields (Classification Research Group, 1978; Vickery, 1960). The FCM is designed to accommodate domain-specific taxonomies. As Spiteri [6](1998) notes in her simplified model for facet analysis, the key is empirical domain analysis to determine which facets best serve particular knowledge communities.

For example, suppose an LLM answers our turmeric query with the claim, "Turmeric's active component curcumin has anti-inflammatory properties and is widely used in Ayurvedic healing in India." We can facet-tag this claim as: Medicine (Main Class) Ayurveda (System) Personality (P): Turmeric (or Curcumin); Matter (M): anti-inflammatory medicinal use; Energy (E): healing process; Space (S): India. These facets map to entries in a faceted knowledge organisation system (e.g., Ayurveda or pharmacology classes). In practice, the validator component will label "Turmeric" under Personality, "anti-inflammatory" under Matter, "healing" under Energy, and so on. Each facet tag then guides us to where in the taxonomy or knowledge graph this claim belongs.

In sum, facets turn opaque text into semantic tags. This resembles earlier work on classification as a formal language [15] . Each sub class label or facet category becomes a meaningful unit bridging human understanding and AI output. By grounding every claim in PMEST terms, we reduce confusion: facets serve as an explicit index that a user or system can inspect, analogous to a call number for an answer.

Planes of Knowledge Creation (Idea, Verbal, Notational)

Ranganathan's three planes of work—Idea, Verbal, Notational—describe how knowledge evolves from thought to representation to classification [4] .It has been reinterpreted here for the LLM context, applying them to both the user's query process and the model's output.

Idea Plane: This is the user's underlying intent or concept. In the example stated above, the user's idea is "curcumin/turmeric used in traditional Indian medicine." It encompasses the subject matter the user has in mind, including any tacit cultural context (Ayurveda, traditional healing). The idea plane is pre-linguistic: it could involve intuitive knowledge about Ayurvedic uses of herbs, not yet formulated as words.

Verbal Plane: This is the actual natural language of the prompt and response. When the user forms the query "What are the uses of turmeric in Indian medicine?", they move their idea onto the verbal plane. The LLM also operates on the verbal plane internally (processing word tokens) and produces textual output on this plane. Because LLMs are sensitive to phrasing, synonyms, and language nuances, the verbal plane influences retrieval and generation.

Notational Plane: In Ranganathan's sense, the notational plane is the classification notation (e.g., class number). This notion is extended here to mean any formal encoding of concepts—classification codes, ontology identifiers, or knowledge graph nodes. After the LLM generates its verbal answer, the notational plane is where the answer is mapped onto structured categories. Concretely, if the LLM mentions "turmeric," a classification overlay would assign it to the notational index for Curcuma longa in a botanical ontology. This mapping may involve multiple domain classifiers.

To illustrate with turmeric: the idea plane is the user's curiosity about turmeric's medicinal uses. The verbal plane is the exact prompt and the LLM's textual answer. The notational plane would assign formal tags like (Main Class: Medicine, System: Ayurveda, P:Turmeric, M:Medicinal plants, E:Treatment, S:India). By explicitly performing this notational step, we anchor the textual answer in a knowledge system. This helps disambiguate the prompt and reveals misalignments: if the LLM drifts into a different interpretation (e.g., discussing turmeric in cosmetics), the notational tags will not match the Ayurvedic class.

By explicitly separating these planes, we enforce a mapping step which Ranganathan advocated. Without it, the LLM's answer floats untethered; with it, each piece of knowledge can be checked against the classification.

Classification as Semantic Language

A classification system is like a controlled language [15] . Each class or facet term carries agreed meaning, just as words do in English. Ranganathan saw classification as a formal language of knowledge. In the approach adopted here, the FCM's taxonomy and the facet labels serve as an artificial semantic layer on top of the LLM. For instance, in the turmeric example, the term "Ayurveda" in the classification (System facet, a particular philosophy) has a precise definition grounded in traditional Indian medicine. When the LLM's output is tagged with that facet, it inherits that precise meaning.

This classification language can also impose constraints. For example, if the knowledge base says "Turmeric belongs to Family Zingiberaceae, used for digestive and anti-inflammatory functions," then an answer claiming "turmeric cures diabetes in two weeks" would violate the facet tag Matter:medical use, because that claim is not recognized in the Ayurveda or pharmacology classes. The validator can flag such a mismatch. In effect, the facets and classes form a semantic net that helps identify contradictions or ungrounded assertions.

Epistemic Hierarchy of Claims

Beyond syntactic and semantic consistency, we also need to assess the epistemic status of each claim. We adopt a seven-level Epistemic Hierarchy (inspired by Ranganathan's intellection/intuition and by evidence grading schemes used in evidence-based medicine) to rate an LLM's answers by certainty and type of knowledge [8,9,16]. Each level represents a different quality and type of supporting evidence:

Level 1 - Opinion/Belief: Unsupported statements with no verifiable evidence. These represent personal views, hunches, or unsubstantiated claims. Example: "Turmeric might be helpful for skin health" (no evidence provided).

Level 2 - Speculation/Hypothesis: Informed conjectures that are testable but not yet confirmed through rigorous study. Example: "Some studies suggest turmeric could reduce inflammation" (vague reference without specifics).

Level 3 - Observation/Anecdote: Empirical claims based on limited data, case reports, or practitioner observations. Example: "Many Ayurvedic practitioners observe benefits of turmeric in wound healing" (anecdotal, not systematically validated).

Level 4 - Correlation/Association: Statistical relationships identified without proven causal mechanisms. Example: "Research shows a correlation between turmeric intake and lower inflammation markers" (association found, causation unclear).

Level 5 - Causal Explanation/Theory: Well-supported causal claims based on controlled experiments or validated theories. Example: "Curcumin in turmeric inhibits inflammatory pathways, as shown in lab experiments" (mechanism understood through experimentation).

Level 6 - Validated Knowledge/Consensus: Widely accepted facts with independent confirmation, often from meta-analyses or expert consensus. Example: "Turmeric's anti-inflammatory effects are recognized by pharmacologists and traditional medicine alike" (broad agreement across sources).

Level 7 - Demonstrated Fact/Proof: Definitive, reproducible knowledge, often quantitative or mathematical, confirmed through rigorous replication. Example: "The molecular pathway by which curcumin reduces interleukin levels is experimentally confirmed" (hard science, reproducible results).

Each LLM output claim is tagged with one of these levels depending on the evidence it cites or can be verified against. For example, if the LLM says "Turmeric cures cancer," that is likely Level 1 or 2 (no evidence), whereas "Clinical trials have shown curcumin reduces inflammatory cytokines" might be Level 5 or 6. Our validator uses these levels to signal confidence and guide user trust. In a medical domain, users might require Level ≥6 ("validated consensus") before acting on advice.

The implementation mechanisms for epistemic scoring are detailed in Appendix B, including fact-checking models, citation verification approaches, confidence scoring based on source agreement, and study quality assessment criteria. A complete reference table mapping each level to evidence types and implementation approaches is provided in Appendix B.

The epistemic hierarchy also structures iterative dialogue. If the answer is at Level 3 or 4 (correlation), the user might follow up with prompts like "Provide an experimental study" to push the response to Level 5 or 6. In this way, repeated prompting and checking can move knowledge up the hierarchy. The LLM and validator thus engage in a staged dialogue of increasing certainty—reminiscent of how scientific knowledge is built from hypothesis to theory to law.

Conceptual Model Proposal

An outline of how to implement the above principles in practice is presented here. Our conceptual model has two main components: (1) a Faceted Classification Module (FCM)—an external knowledge organisation system that encodes facets and classes, and (2) an LLM Validator/Classifier that operates alongside the LLM. In use, a user submits a query (e.g., about turmeric), the LLM generates an answer, and the validator analyses the answer claim-by-claim using modern NLP and machine learning techniques.

Faceted Classification Module (FCM)

The FCM is essentially a knowledge organisation system built on Ranganathan's principles, extended with CRG insights for domain-specific flexibility. It could be an enriched Colon Classification schedule, an ontology, or a knowledge graph annotated with PMEST facets. It contains entries for subjects, entities, and facts, each annotated by facets. Key elements include:

Facet Hierarchy: The FCM organises entities with facet labels. For example, one entry might be "Turmeric (Curcuma longa)" with (P:Turmeric/Spice/Herb, M:Curcumin/Essential Oil, E:Medicinal use, S:India, System:Ayurveda). These tags are linked to broader classes (e.g., Plant drugs, Inflammation treatment).

Core Axioms: Fundamental truths or axioms (intuition-based knowledge) can be encoded as special classes. For instance, Ranganathan's notion of "axioms" might correspond to base scientific facts accepted as given. These are foundational elements in the knowledge structure.

Growth Mechanism: In keeping with Ranganathan's "ever-growing universe of knowledge" , the FCM must be extensible. If a novel claim arises about a natural element having properties similar to turmeric , the system can create an isolate to place the element close by in relation to turmeric , without disrupting established categories. Over time, new findings can be integrated smoothly.

Epistemic Labels: Optionally, each FCM entry might include an epistemic level (1-7) indicating its certainty or consensus status. For example, a fact derived from a well-known pharmacology textbook would be Level 7, while a folk claim might be Level 2. This way, the FCM not only classifies content but also stores its confidence.

In practice, building the FCM might involve merging existing taxonomies and knowledge graphs with facet annotations. For our turmeric example, the FCM would include an entry such as: Turmeric—(P:Turmeric, M:Anti-inflammatory, E:Digestive healing, S:India, System:Ayurvedic). It might also have entries for related concepts (e.g., Curcumin, Vedic medicine, Zingiberaceae family). These provide the "ground truth" against which answers are checked.

LLM with Classification Overlay: Validation Pipeline

The LLM operates as usual to generate text. The innovation is the Validator, a pipeline that post-processes the LLM's output using state-of-the-art NLP and machine learning techniques. After the LLM answers the query, we execute an eight-step process:

Step 1: Claim Segmentation - Split the answer into individual claims or facts using dependency parsers and clause segmentation models.

Step 2: Entity Recognition (Personality Facet) - Identify primary entities using transformer-based Named Entity Recognition models and link them to the FCM through ontology alignment.

Step 3: Matter Facet Identification - Extract attributes, properties, and qualities through modifier extraction and multi-label classification.

Step 4: Energy Facet Identification - Identify actions, processes, and transformations using relation extraction models and verb mapping.

Step 5: Space and Time Facets - Tag geographic and temporal context using geospatial and temporal tagging tools.

Step 6: Category Lookup and Consistency Checking - Verify claims against the FCM knowledge base using graph queries and ontology-based logical constraints.

Step 7: Epistemic Level Assignment - Assign certainty levels (1-7) based on evidence retrieval and fact-checking models.

Step 8: Feedback Loop (Optional) - If claims fail validation or receive low epistemic scoring, engage the LLM in iterative refinement by prompting for evidence or rephrasing.

The specific NLP models, ML approaches, and technical implementation details for each step are provided in Appendix A, including tool selections (SciSpacy, BioBERT, IndicBERT for NER; OpenIE, DyGIE++ for relation extraction; Sentence-BERT with FAISS for ontology alignment; SPARQL, Cypher, SHACL for graph queries), parameter configurations, and workflow specifications.

Thus, the LLM's text becomes annotated with facet tags, category codes, and confidence levels. The final output to the user includes not just the answer, but a facet breakdown and a certainty rating. This dual output is like a library catalog card attached to the answer, saying exactly where it fits in the knowledge universe and how well supported it is.

Classification Validator in Action: Turmeric Example

Consider the example query "What are the uses of turmeric in Indian medicine?" The LLM might respond: "Turmeric contains curcumin which has antioxidant properties and is used to treat arthritis and digestive issues in Ayurveda." Our validator would process as follows:

Claims Extracted:

1. "Turmeric contains curcumin."
2. "Curcumin has antioxidant properties."
3. "(Turmeric) is used to treat arthritis in Ayurveda."

4. "(Turmeric) is used to treat digestive issues in Ayurveda."

Facet Tags:
- Claim 1: P:Turmeric, M:Curcumin (Matter), E:contains
- Claim 2: P:Curcumin, M:antioxidant property, E:has
- Claim 3: P:Turmeric, M:Arthritis, System: India/Ayurveda, E:treatment
- Claim 4: P:Turmeric, M:Digestive Issues, System: India/Ayurveda, E:treatment

Category Lookup: The validator checks the FCM: it finds classes like "Turmeric in Ayurvedic medicine" and "Curcumin in plant extracts." It confirms that turmeric is indeed classified under spice medicinal uses and that antioxidant and digestive uses are recognized.

Consistency Check: All four claims align with FCM knowledge (arthritis and digestion are known Ayurvedic uses of turmeric). If the LLM had said "turmeric cures diabetes completely in two weeks," the validator would fail to find a class supporting this extreme claim and would flag it as inconsistent or unsupported.

Epistemic Level: Given these facts are well-established (multiple studies support anti-inflammatory and digestive benefits of curcumin), each claim would be Level 6 (validated consensus). The system provides citations to meta-analyses or authoritative pharmacology texts. If any claim were dubious or lacked supporting evidence, it would receive a lower score, alerting the user.

This walkthrough demonstrates how each piece of output is tagged with PMEST facets and evaluated. Crucially, the user receives a richer answer: not just the raw text, but its breakdown into concepts and transparency into the reasoning. The following schematic representation demonstrates the stages of post-processing a LLM response to generate a hallucinator index. The response of the LLM is passed through a 2 stage pipeline - stage 1 for facet-based validation, and stage 2 for knowledge-base consistency. The following schematic is an illustration

Integration with Graph-Based and RAG Approaches

Many contemporary systems aim to bridge LLMs with knowledge graphs (KGs). Most notably, Microsoft Research's GraphRAG constructs a knowledge graph from input texts and uses it to augment LLM prompts (Edge et al., 2024). GraphRAG follows a Retrieval-Augmented Generation (RAG) approach: it indexes a corpus into a graph (entities and relations), clusters entities into communities, generates summaries, and at query time

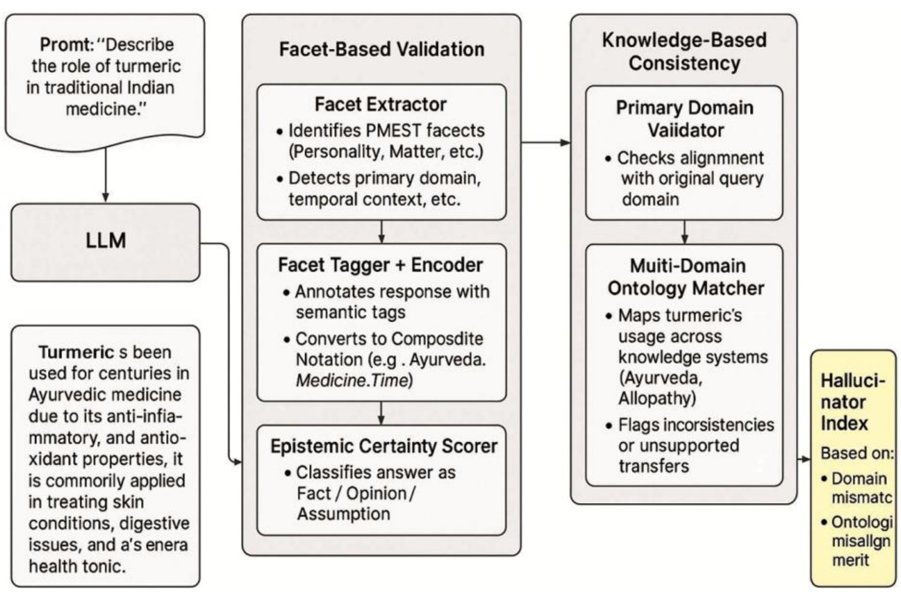

Figure 1. Schematic of Facet Classification System to LLM response to determine a Hallucinator Index for a LLM response. In summary, the conceptual framework embeds Ranganathan's classification both before (pre-structuring the knowledge) and after (post-validating the answer) LLM generation. It transforms the LLM from a black-box text generator into a component within a structured knowledge- organizing workflow. The rest of the article discusses the implications and benefits of this approach.

feeds relevant subgraphs into the LLM context. This retrieval-based pipeline focuses on correspondence: the graph provides factual snippets that the LLM can cite, reducing hallucination.

GraphRAG vs. Baseline RAG

As Microsoft's documentation notes, standard vector-search RAG "struggles to connect the dots" across disparate information and performs poorly on holistic questions [10] . GraphRAG addresses this by building a global graph structure; it reports substantial improvements on multi-hop and summary questions. Similarly, LangChain and Neo4j tutorials demonstrate hybrid retrieval: they construct a KG via an LLM and then combine graph queries with vector and keyword search [17] . This hybrid approach "achieves the best of both worlds," capturing structured relationships and unstructured context. An IBM tutorial echoes this, describing Graph RAG as "emerging as a powerful technique" that leverages the structured nature of graphs to give greater context than vector search alone [18] .

Comparison to Our Classification Model

GraphRAG and similar tools primarily act before and during answer generation—they equip the LLM with curated knowledge. Our model acts after generation, as a verification and organization layer. In GraphRAG's terms, they rely on retrieval fidelity: ensuring the answer reflects retrieved documents. We rely on a coherence theory of truth: ensuring the answer coheres internally and with the classification structure.

Importantly, the two approaches are complementary, not mutually exclusive. For example, our facets could serve as intelligent filters in a KG-RAG pipeline. When GraphRAG handles a query, it could use facet tags from the query or preliminary answer to select which subgraph to search (e.g., only retrieve nodes with P:Turmeric, M:Medicine for our example). Conversely, the FCM could incorporate knowledge graphs: instead of a pure faceted schedule, the FCM might be implemented as a KG with PMEST annotations on nodes. After GraphRAG produces an answer, our validator could then audit it: even if retrieval was successful, we would still perform a cross-facet consistency check.

An illustrative difference: GraphRAG might take the turmeric query and return passages from Ayurvedic texts, which the LLM paraphrases. Our model, by contrast, would tag and check each paraphrased claim. If GraphRAG's answer implies "turmeric cures X" and our facets indicate no class exists for that claim, the validator flags it. We can imagine a hybrid system: GraphRAG for knowledge retrieval and our classification layer for integrity checking.

Hybrid Systems in Practice

Emerging tools combine RAG with graphs. For example, LangChain's libraries now include graph-qa chains that connect LlamaIndex or Neo4j graphs to LLMs [17]. IBM's tutorial shows how to pair Meta's Llama-3 with a Memgraph KG to answer questions: the LLM both creates and queries the graph [18]. These pipelines typically improve over text-only RAG by ensuring retrieved facts are structured and up-to-date. Peng et al. [12] (2023) demonstrate that external knowledge and automated feedback loops significantly improve LLM factual accuracy.

However, most KG-augmented RAG systems are designed for factual coverage, not cultural nuance. By contrast, our faceted approach emphasizes interpretability and cultural context. For instance, by creating an "Ayurveda" facet class in the FCM, we ensure the LLM respects indigenous categories. GraphRAG, while powerful, is not specifically designed for cultural taxonomy. Our model can embed culturally rooted knowledge systems explicitly: e.g., introducing facets for Sanskrit terms or local classification schemes. In summary, retrieval-based tools excel at data gathering and breadth, whereas our classification model adds a validation/organization overlay. Together, they point toward a holistic LLM architecture: one that combines retrieval fidelity (for factual grounding) and epistemic structure (for coherence and explainability) in a feedback loop.

Implementation Roadmap and Evaluation

To realise this model, we propose an implementation architecture and evaluation plan with the following elements:

System Architecture

The FCM may be implemented as a graph database (e.g., Neo4j, Amazon Neptune) or an RDF triple-store with PMEST annotations. It can integrate multiple existing knowledge bases: for medicine, use UMLS/MeSH; for general facts, use Wikidata or DBpedia; for cultural domains, incorporate domain-specific ontologies (e.g., Ayurveda ontologies or ethnobotany databases). The LLM (e.g., GPT-4, Claude) runs as a service (via API) that generates answers. The Validator is a post-processing pipeline implemented as microservices that can be orchestrated using frameworks like LangChain or LlamaIndex.

These components communicate via APIs. Scalability is addressed by sharding the FCM by domain and indexing common facets. The FCM graph can be updated continuously without retraining the LLM; it can also incorporate crowdsourced edits or expert input for new facets.

Datasets and Knowledge Sources

To build and test the FCM, we will use structured data sources such as Wikidata, UMLS/MeSH, Gene Ontology, and specialized taxonomies for Ayurveda or local knowledge. For example, Wikidata's turmeric entry and its subclass relationships provide facet candidates; Ayurvedic pharmacopeias can seed cultural categories. Training data for the Validator (facet tagging) can be semi-automatically generated by linking Wikipedia or domain corpora to the PMEST schema. Evaluation will use benchmark datasets: fact-checking tasks like FEVER (with Wikipedia-backed claims), SciFact (scientific claim verification), and domain-specific QA sets. We also plan a custom test set of culturally-grounded queries to measure the system's handling of indigenous knowledge.

Performance Metrics

The system's efficacy will be measured along several dimensions:

- **Factual accuracy:** Precision/recall of correct claims vs hallucinations in final answers
- **Facet tagging accuracy:** How often the validator assigns the correct facet to each concept, compared to annotated data
- **Consistency detection rate:** How often contradictions in LLM output are successfully flagged
- **User trust/confidence metrics:** Via surveys or simulated user tests, measuring whether providing the facet breakdown and certainty levels increases perceived reliability

We will compare against baselines: the raw LLM output, and LLMs augmented with conventional RAG (vector retrieval) without classification. We can adopt ROUGE or BERTScore metrics for answer relevance [19,13] and design custom "consistency" metrics that penalize unsupported assertions. Reduction in hallucination rate is a key metric, following Chelli et al [2]. (2024).

Scalability Considerations

The FCM graph can grow large, but query performance can be maintained by indexing and caching. Facet tagging using transformer models can be parallelized on GPUs, making real-time response feasible (sub-second per claim). Since each query answer typically contains a handful of claims, the overhead is modest. The graph database can handle millions of nodes, and partitioning by facet can improve throughput. The system can be deployed in a modular cloud environment, allowing horizontal scaling of the Validator pipeline. Initial estimates suggest the validator adds on the order of a few seconds per query, acceptable for high-value applications. Future work may use approximate embeddings to speed up facet matching, or federated queries across multiple KGs.

Culturally Rooted AI and Future Work

A core motivation for our model is trustworthiness and inclusivity. By framing an LLM's output within a transparent faceted system, we make the AI's "reasoning" traceable. Users can see why information is accepted or flagged. This aligns with Explainable AI goals [20] . For instance, a doctor querying the system might see not just that "turmeric treats inflammation," but also the classification path: "Medicine > Pharmacology > Herbal Remedies > Turmeric:Anti-inflammatory", along with a high confidence level (Level 6). This bibliographic context builds confidence.

Equally important is cultural context. Ranganathan's system was developed in India and aimed at a diverse readership. We borrow that spirit to respect indigenous knowledge. In practice, our FCM would be co-designed with cultural experts. For example, Ayurveda has concepts like doshas and uses Sanskrit categories. We could create a facet or class "Ayurveda" with sub-facets (Vata, Pitta, Kapha) and use those in tagging. If a query is about turmeric in Indian medicine, the FCM might recognize Ayurveda as a valid context; if the LLM instead defaults to Western pharmaceutical terms, the facet tags will not align, signaling a mismatch. This could prevent cultural erasure in AI answers.

This approach resonates with initiatives in India to integrate Indian Knowledge Systems (IKS) into AI and education. A faceted validator could, for example, assign the claim "turmeric cleanses toxins" to a traditional category of Ayurveda and highlight it as validated by community texts, rather than by biomedical sources. It could even output the Sanskrit or vernacular term as a Personality facet. In our view, an LLM that includes classification is inherently more inclusive: it lets multiple knowledge traditions coexist in the same framework, rather than the model defaulting to its (often Western-centric) training data.

References

1 Ranganathan, S.R. (1967), *Prolegomena to Library Classification*, Asia Publishing House, Bombay, 3rd ed.

2 Chelli, M., Descamps, J., Lavoué, V., Trojani, C., Azar, M., Deckert, M., Raynier, J.L., Clowez, G., Boileau, P. and Ruetsch-Chelli, C. (2024), "Hallucination rates and reference accuracy of ChatGPT and Bard for systematic reviews: comparative analysis", Journal of Medical Internet Research, Vol. 26, e53164. https://doi.org/10.2196/53164.

3 Mahaut, F., Mokady, R., Aharoni, E. and Berant, J. (2024), "Factual Confidence of LLMs: on Reliability and Robustness of Current Estimators", in Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024),pp. 4554-4570.

4 Dousa, T.M. (2019), "Intellection and intuition: On the epistemology of S.R. Ranganathan", *Knowledge Organization*, Vol. 46 No. 3, pp. 149-174.

5 Vickery, B.C. (1960), *Faceted Classification Schemes*, Rutgers University Graduate School of Library Service, New Brunswick, NJ.

6 Spiteri, L.F. (1998), "A simplified model for facet analysis", *Canadian Journal of Information and Library Science*, Vol. 23 No. 4, pp. 1-30.

7 Tennis, J.T. (2008), "Form, function, and the domains of classification", in *Cultural Frames of Knowledge*, Ergon Verlag, Würzburg, pp. 197-205.

8 Guyatt, G.H., Oxman, A.D., Vist, G.E., Kunz, R., Falck-Ytter, Y., Alonso-Coello, P. and Schünemann, H.J. (2008), "GRADE: An emerging consensus on rating quality of evidence and strength of recommendations", *BMJ*, Vol. 336 No. 7650, pp. 924-926.

9 Oxford Centre for Evidence-Based Medicine (2011), "Levels of evidence", available at: https://www.cebm.ox.ac.uk/resources/levels-of-evidence (accessed 15 October 2025).

10 Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S. and Larson, J. (2024), "From local to global: A graph RAG approach to query-focused summarization", *arXiv preprint*, arXiv:2404.16130.

11 Hjørland, B. (2014), "Is facet analysis based on rationalism? A discussion of Satija (1992), Tennis (2008), Herre (2013), Mazzocchi (2013b), and Dousa & Ibekwe-SanJuan (2014)", *Knowledge Organization*, Vol. 41 No. 5, pp. 369-376.

12 Peng, B., Galley, M., He, P., Cheng, H., Xie, Y., Hu, Y., Huang, Q., Liden, L., Yu, Z., Chen, W. and Gao, J. (2023), "Check your facts and try again: Improving large language models with external knowledge and automated feedback", *arXiv preprint*, arXiv:2302.12813.

13 Zheng, D., Lapata, M. and Pan, J.Z. (2024), "Large language models as reliable knowledge bases?", *arXiv preprint*, arXiv:2407.13578.

14 Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J. and Wu, X. (2024), "Unifying large language models and knowledge graphs: A roadmap", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 36 No. 7, pp. 3580-3599.

15 van den Heuvel, C. and Smiraglia, R.P. (2011), "Interactions between elementary structures in universes of knowledge", in Dousa, T.M. (Ed.), *Epistemology in Knowledge Organization*, K. G. Saur, Munich, pp. 123-139.

16 Sackett, D.L., Rosenberg, W.M., Gray, J.A., Haynes, R.B. and Richardson, W.S. (1996), "Evidence based medicine: What it is and what it isn't", *BMJ*, Vol. 312 No. 7023, pp. 71-72.

17 LangChain (2024), "Enhancing RAG-based applications' accuracy by constructing and leveraging knowledge graphs", *LangChain Blog*, available at: https://blog.langchain.com/enhancing-rag-based-applications-accuracy-by-constructing-and-leveraging-knowledge-graphs/ (accessed 3 June 2025).

18 IBM (2025), "Graph RAG: Combining LLMs with knowledge graphs", available at: https://www.ibm.com/think/tutorials/knowledge-graph-rag/ (accessed 28 June 2025).

19 Lin, C.-Y. (2004), "ROUGE: A package for automatic evaluation of summaries", in *Proceedings of Workshop on Text Summarization (WAS 2004)*, Barcelona, Spain, pp. 74-81.

20 Bilal, A., Ebert, D. and Lin, B. (2025), "LLMs for explainable AI: A comprehensive survey", *arXiv preprint*, arXiv:2504.00125v1.

Additional Readings

1 Classification Research Group (1978), "The need for a faceted classification as the basis of all methods of information retrieval", in Maltby, A. (Ed.), *Classification in the 1970s: A Second Look*, Clive Bingley, London, pp. 87-104.

2 Honnibal, M. and Montani, I. (2017), "spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing", available at: https://spacy.io (accessed 15 October 2025).

3 Kakwani, D., Kunchukuttan, A., Golla, S., Gokul, N.C., Bhattacharyya, A., Khapra, M.M. and Kumar, P. (2020), "IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages", in *Findings of EMNLP 2020*, pp. 4948-4961.

4 Knublauch, H. and Kontokostas, D. (2017), "Shapes Constraint Language (SHACL)", W3C Recommendation, available at: https://www.w3.org/TR/shacl/ (accessed 15 October 2025).

5 Neumann, M., King, D., Beltagy, I. and Ammar, W. (2019), "ScispaCy: Fast and robust models for biomedical natural language processing", in *Proceedings of the 18th BioNLP Workshop and Shared Task*, Association for Computational Linguistics, Florence, pp. 319-327.

6 Reimers, N. and Gurevych, I. (2019), "Sentence-BERT: Sentence embeddings using Siamese BERT-networks", in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3982-3992.
7 Strötgen, J. and Gertz, M. (2013), "Multilingual and cross-domain temporal tagging", *Language Resources and Evaluation*, Vol. 47 No. 2, pp. 269-298.
8 Wadden, D., Wennberg, U., Luan, Y. and Hajishirzi, H. (2019), "Entity, relation, and event extraction with contextualized span representations", in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*, pp. 5784-5789.
9 Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q. and Artzi, Y. (2019), "BERTScore: Evaluating text generation with BERT", *arXiv preprint*, arXiv:1904.09675.

APPENDIX A: Technical Implementation Details for Faceted Classification Module and Validator

This appendix provides comprehensive technical specifications for implementing the Faceted Classification Module (FCM) and Classification Validator, addressing the peer review feedback regarding specific NLP techniques and machine learning approaches for automatic facet identification and tagging.

A.1 Claim Segmentation Module

Objective: Decompose LLM-generated text into atomic, independently verifiable claims.

Technologies:

- **Dependency Parsers:** spaCy (Honnibal and Montani, 2017) with en_core_web_trf transformer model for English; Stanford CoreNLP for multi-language support
- **Clause Segmentation:** Rule-based extraction using dependency relations (nsubj, dobj, ccomp, xcomp)
- **Coreference Resolution:** Neural coref models (e.g., AllenNLP's coref-spanbert) to resolve pronouns and ensure each claim is self-contained

Implementation Workflow:

1. Parse input text using spaCy dependency parser
2. Identify sentence boundaries and clause structures
3. Extract independent clauses as separate claims
4. Resolve coreferences to ensure atomic claims contain complete information
5. Output: List of independent claim strings

Example:

- Input: "Turmeric contains curcumin which has anti-inflammatory properties."
- Output: ["Turmeric contains curcumin", "Curcumin has anti-inflammatory properties"]

A.2 Named Entity Recognition and Personality Facet Detection

Objective: Identify primary entities (Personality facet) in each claim with domain-specific precision.

Models:

Biomedical Domain:

- **SciSpacy** (Neumann et al., 2019): en_ner_bc5cdr_md model for disease/chemical recognition
- **BioBERT**: Fine-tuned on PubMed abstracts for medical terminology
- **BERN2**: Biomedical entity recognition service combining multiple NER models

Cultural/Multilingual:

- **IndicBERT** (Kakwani et al., 2020): For Hindi, Tamil, Sanskrit, and other Indian language terms
- **mBERT**: Multilingual BERT for cross-lingual entity recognition
- **Custom Ayurveda NER**: Fine-tuned model on digitized Ayurvedic texts

Implementation Workflow:

1. Apply domain-appropriate NER model to each claim
2. Extract entities with confidence scores

3. Perform entity linking using Sentence-BERT embeddings
4. Map entities to FCM using FAISS similarity search (cosine similarity threshold ≥ 0.85)
5. Assign Personality facet tags

Entity Linking Process:
- Encode claim entities and FCM entries using Sentence-BERT (Reimers and Gurevych, 2019)
- Build FAISS index of FCM entity embeddings
- Query index with claim entity embeddings
- Return top-k matches (k=5) with similarity scores
- Human-in-the-loop verification for ambiguous matches (<0.85 similarity)

Output Format:
```
{
  "claim": "Turmeric contains curcumin",
  "entities": [
    {"text": "Turmeric", "type": "PLANT", "fcm_id": "P:Curcuma_longa", "confidence": 0.94},
    {"text": "curcumin", "type": "CHEMICAL", "fcm_id": "P:Curcumin", "confidence": 0.97}
  ]
}
```

A.3 Matter Facet Extraction

Objective: Identify attributes, properties, qualities, and characteristics associated with entities.

Techniques:

Modifier Extraction:
- Dependency parsing to extract adjectival modifiers (amod), noun compounds (compound), and prepositional modifiers (nmod)
- Pattern matching: [Entity] + [is/has/contains] + [Attribute]

Multi-label Classification:
- Fine-tuned DeBERTa-v3 on domain-specific attribute taxonomies
- Training data: Medical ontologies (UMLS semantic types), material properties databases
- Output: Probability distribution over attribute categories (biological_function, chemical_property, physical_state, medicinal_use, etc.)

Ontology Alignment:
- Map extracted attributes to Matter facet categories in FCM
- Use WordNet synsets for semantic expansion
- ChEBI ontology for chemical properties
- GO (Gene Ontology) for biological functions

Implementation:
```
# Pseudocode

def extract_matter_facets(claim, entities):
    modifiers = extract_dependencies(claim, rel_types=['amod', 'compound'])
    attributes = classify_attributes(modifiers, model=deberta_classifier)
    matter_facets = align_to_fcm(attributes, facet_type='Matter')
    return matter_facets
```

Example:
- Claim: "Curcumin has anti-inflammatory properties"
- Extracted: "anti-inflammatory" (amod relation)

- Classification: {biological_function: 0.92, medicinal_use: 0.85}
- FCM Mapping: M:Anti-inflammatory_agent, M:Medicinal_property

A.4 Energy Facet Identification
Objective: Detect actions, processes, transformations, and functional relationships.

Relation Extraction Tools:
OpenIE (Open Information Extraction):
- Stanford OpenIE 5.0 for triplet extraction (subject, predicate, object)
- Identifies action verbs and their arguments
- Lightweight, suitable for real-time processing

DyGIE++ (Wadden et al., 2019):
- Deep learning model for entity, relation, and event extraction
- Pre-trained on scientific literature (SciERC dataset)
- Better performance on complex, domain-specific relations
- Outputs typed relations with confidence scores

Verb Semantic Analysis:
- VerbNet classification to categorize actions (creation, transformation, motion, cognitive, etc.)
- FrameNet semantic frames for process understanding
- PropBank for argument structure analysis

Implementation Workflow:
1. Apply OpenIE to extract (subject, verb, object) triplets
2. Use DyGIE++ for complex relation extraction in specialized domains
3. Classify verbs using VerbNet categories
4. Map action/process to Energy facet taxonomy in FCM
5. Validate that Energy facet aligns with entity types (Personality) and attributes (Matter)

Example:
- Claim: "Turmeric is used to treat arthritis"
- OpenIE output: (Turmeric, is used to treat, arthritis)
- Verb classification: "treat" → VerbNet class: remedy-45.7 (medical treatment)
- FCM mapping: E:Treatment, E:Therapeutic_process

A.5 Space and Time Facet Tagging
Objective: Capture geographic, cultural, and temporal context.
Geospatial NER:
- spaCy GPE (GeoPolitical Entity) recognition for countries, cities, regions
- GeoNames database integration for disambiguation and hierarchical location data
- Cultural context recognition: Identify tradition names (Ayurveda, TCM, Western medicine) as spatial context

Temporal Tagging:
- **HeidelTime** (Strötgen and Gertz, 2013): Multilingual temporal tagger
- Extracts explicit dates, periods, durations
- Normalizes temporal expressions to ISO 8601 format
- **SUTime**: Stanford temporal tagger for complex temporal reasoning
- **Historical period recognition**: Custom classifier for era identification (Ancient, Medieval, Modern, Traditional, Contemporary)

Cultural Context Module:
- Knowledge system identification (Ayurveda, Unani, Siddha, Biomedicine)
- Tradition-specific terminology detection (dosha, qi, humors)
- Assigns Space facet for knowledge tradition alongside geographic location

Implementation:
```
def extract_space_time_facets(claim):
    # Spatial
    locations = spacy_ner(claim, types=['GPE', 'LOC'])
    geonames_data = geonames_lookup(locations)
    traditions = identify_knowledge_system(claim)


    # Temporal
    temporal_expr = heideltime.extract(claim)
     periods = classify_historical_period(claim)
    space_facets = {
        'geographic': geonames_data,
        'cultural': traditions
    }

    time_facets = {
        'explicit': temporal_expr,
        'periods': periods
    }

    return space_facets, time_facets
```

Example:
- Claim: "Turmeric is used in Ayurvedic healing in India"
- Space: S:India (geographic), S:Ayurveda (knowledge tradition)
- Time: T:Traditional_period (inferred from "Ayurvedic")

A.6 Category Lookup and Consistency Checking

Objective: Verify extracted facets against FCM and detect logical inconsistencies.

Graph Query Technologies:

RDF Stores (for ontology-based FCM):
- **Apache Jena Fuseki**: SPARQL endpoint for RDF triple stores
- **Virtuoso**: High-performance RDF storage
- Query language: SPARQL 1.1 with federated queries

Property Graphs (for networked FCM):
- **Neo4j**: Graph database with Cypher query language
- **Amazon Neptune**: Managed graph database supporting both RDF and property graphs
- **TigerGraph**: Scalable graph database for complex path queries

Consistency Validation:

SHACL (Shapes Constraint Language) (Knublauch and Kontokostas, 2017):
- W3C standard for RDF graph validation
- Define shape constraints for valid facet combinations
- Automated validation reports with violation details

OWL Reasoning:

- Use Pellet or HermiT reasoners for logical inference
- Detect contradictions through description logic
- Infer implicit relationships from explicit facets

Cross-facet Coherence Rules:

```
# SPARQL query to check if claimed treatment aligns with known uses

SELECT ?claim ?facet_P ?facet_E ?known_use
WHERE {
  ?claim :hasPersonalityFacet ?facet_P ;
         :hasEnergyFacet ?facet_E .
  ?fcm_entry :represents ?facet_P ;
             :hasKnownUse ?known_use .
  FILTER (?facet_E NOT IN (?known_use))
}
```

Implementation Workflow:

1. Construct PMEST signature from extracted facets
2. Query FCM graph for matching entries
3. Apply SHACL shapes for structural validation
4. Run OWL reasoner for logical consistency
5. Check cross-facet rules (e.g., if P:Turmeric and E:Antibiotic, flag inconsistency)
6. Generate consistency report with confidence scores

Output:

```
{
  "claim": "Turmeric treats arthritis",
  "facets": {
    "P": "Turmeric",
    "M": "Anti-inflammatory",
    "E": "Treatment",
    "S": "Ayurveda",
    "T": "Traditional"
  },

  "fcm_match": true,

  "consistency": {
    "valid": true,
    "confidence": 0.89,
    "supporting_entries": ["FCM:Ayur_Turmeric_001", "FCM:Pharm_Curcumin_045"],
    "violations": []
  }
}
```

A.7 Epistemic Level Assignment

Objective: Assign evidence-based certainty levels (1-7) to each claim.

Fact-Checking Models:

SciFact (Wadden et al., 2019):

- Dataset: 1.4K scientific claims with expert annotations
- Model: Fine-tuned RoBERTa for claim verification
- Training objective: 3-way classification (SUPPORTS, REFUTES, NOT_ENOUGH_INFO)

FEVER (Fact Extraction and VERification):
- Large-scale dataset with Wikipedia evidence
- Models: BERT, DeBERTa fine-tuned for claim-evidence matching
- Pipeline: Document retrieval → Sentence selection → Claim verification

Citation Verification:
- **Semantic Scholar API**: Retrieve paper metadata and citations
- **PubMed API**: Medical literature search and validation
- **CrossRef**: DOI resolution and citation network analysis

Evidence Quality Assessment:
```
def assess_evidence_quality(claim, sources):
    quality_scores = []
    for source in sources:
        score = {
            'source_type': classify_source_type(source),  # journal, textbook, blog
            'study_design': identify_study_design(source),  # RCT, cohort, case report
            'citation_count': get_citation_count(source),
            'journal_impact': get_journal_metrics(source),
            'replication_status': check_replication_studies(source)
        }

        quality_scores.append(calculate_quality_score(score))

    return aggregate_evidence_quality(quality_scores)
```

Epistemic Level Mapping Algorithm:
1. Retrieve supporting evidence from external sources
2. Count number of supporting/refuting sources
3. Assess quality of each source (peer-reviewed journal > preprint > blog)
4. Evaluate study design (meta-analysis > RCT > observational > anecdote)
5. Check for consensus across sources
6. Map to epistemic levels:

- Level 1: No sources, or only opinion pieces
- Level 2: Hypothesis stated but no studies
- Level 3: Case reports or anecdotal evidence
- Level 4: Observational studies showing correlation
- Level 5: Experimental studies with causal evidence
- Level 6: Multiple high-quality studies + meta-analyses
- Level 7: Replicated experiments with quantitative proof

Confidence Scoring:
- Inter-source agreement (Fleiss' kappa for multiple sources)
- Temporal consistency (recent vs. outdated evidence)
- Cross-cultural validation (evidence from multiple knowledge traditions)

A.8 Feedback Loop and Iterative Refinement

Objective: Enable LLM to improve responses based on validation feedback.

Prompt Engineering:

```
def generate_refinement_prompt(claim, validation_result):
    if validation_result['consistency'] == False:
        return f"The claim '{claim}' appears inconsistent with known facts. \
                Can you provide sources or revise the claim?"

    if validation_result['epistemic_level'] < 4:
        return f"The claim '{claim}' has low evidence support (Level {level}). \
                Can you provide experimental studies or higher-quality sources?"

    return None  # No refinement needed
```

Implementation Workflow:

1. Validator identifies low-confidence or inconsistent claims
2. Generate targeted follow-up prompts
3. Submit refined prompt to LLM
4. Re-run validation pipeline on new response
5. Track improvement metrics (epistemic level increase, consistency improvement)
6. Terminate after N iterations or when threshold met

Metrics:

- Evidence level progression (e.g., Level 2 → Level 5)
- Source quality improvement
- Consistency score change
- User satisfaction ratings

APPENDIX B: Detailed Epistemic Hierarchy with Implementation Guidelines

This appendix provides comprehensive definitions, concrete examples, and implementation approaches for each of the seven epistemic levels. We propose one implementation approach that has to be rigorously validated with more data points. .

B.1 Level 1 - Opinion/Belief

Definition: Unsupported personal views, beliefs, or hunches with no verifiable evidence base. These statements reflect subjective judgment without grounding in empirical observation or expert consensus.

Characteristics:

- No cited sources or references
- Use of subjective language ("I think," "might," "could be")
- Absence of empirical data or theoretical framework
- Often based on individual experience or cultural belief

Examples:

Turmeric Domain:

- "Turmeric might be helpful for skin health."
- "I believe turmeric makes food taste better."
- "Some people say turmeric is good for you."

Other Domains:

- "Electric cars are probably the future of transportation." (no evidence)
- "Classical music seems to help plants grow." (folk belief)
- "Dark matter might not exist." (unfounded speculation)

Detection Criteria:

- Absence of citations or supporting documentation
- Hedging language patterns (might, could, possibly, perhaps)
- No mention of studies, data, or expert sources
- Claims that contradict established knowledge without justification

Implementation:

```
def detect_level_1(claim, sources):
    hedging_keywords = ['might', 'could', 'possibly', 'perhaps', 'maybe', 'probably']
    subjective_markers = ['I think', 'I believe', 'some say', 'it seems']
    has_hedging = any(keyword in claim.lower() for keyword in hedging_keywords)
    has_subjective = any(marker in claim.lower() for marker in subjective_markers)
    has_sources = len(sources) > 0

    if (has_hedging or has_subjective) and not has_sources:
        return True, "Opinion/Belief - No supporting evidence"
    return False, None
```

B.2 Level 2 - Speculation/Hypothesis

Definition: Informed conjectures or testable hypotheses that have been proposed but not yet confirmed through rigorous empirical investigation. These represent educated guesses based on preliminary observations or theoretical reasoning.

Characteristics:

- Testable propositions
- Based on initial observations or logical inference
- Lack of rigorous experimental validation
- May cite preliminary or pilot studies
- Often framed as research questions

Examples:

Turmeric Domain:

- "Some studies suggest turmeric could reduce inflammation." (vague, no specific studies cited)
- "Researchers hypothesize that curcumin may modulate immune function."
- "Initial experiments indicate potential anti-cancer properties."

Other Domains:

- "Scientists speculate that dark energy drives cosmic expansion." (theoretical but unconfirmed)
- "Early data suggest a possible link between social media use and anxiety."
- "Preliminary findings hint at a genetic component to intelligence."

Detection Criteria:

- Use of speculative language ("suggest," "hypothesize," "may," "potential")
- Reference to "preliminary," "pilot," or "initial" studies
- Absence of definitive experimental confirmation
- Claims presented as questions or possibilities rather than facts

Implementation:

```
def detect_level_2(claim, sources):
    speculation_keywords = ['suggest', 'hypothesize', 'may', 'potential', 'possible', 'preliminary']
    has_speculation = any(keyword in claim.lower() for keyword in speculation_keywords)
    has_weak_sources = any('preliminary' in s.title.lower() or
                           'pilot' in s.title.lower() or
                           'abstract' in s.type for s in sources)
    strong_evidence = any(s.type == 'randomized_trial' or
                          s.type == 'meta_analysis' for s in sources)
    if has_speculation and has_weak_sources and not strong_evidence:
        return True, "Speculation/Hypothesis - Preliminary evidence only"
    return False, None
```

B.3 Level 3 - Observation/Anecdote

Definition: Empirical claims based on limited data, practitioner reports, case studies, or observational evidence without systematic validation. These represent real-world observations but lack rigorous controls or large sample sizes.

Characteristics:

- Based on direct observation or experience
- Limited sample size (single case or small group)
- Lack of experimental controls
- May include traditional/indigenous knowledge based on generations of practice
- Practitioner testimonials or clinical observations

Examples:

Turmeric Domain:

- "Many Ayurvedic practitioners observe benefits of turmeric in wound healing."
- "Case reports document patients experiencing pain relief after turmeric supplementation."
- "Traditional healers in India have used turmeric for digestive issues for centuries."

Other Domains:

- "A patient with severe migraines reported improvement after removing gluten from their diet." (case report)
- "Indigenous communities have long observed that certain plants repel mosquitoes."
- "Astronomers noticed unusual radio signals from a distant galaxy."

Detection Criteria:

- References to case reports, case series, or practitioner experience
- Mention of traditional knowledge or indigenous practice
- Observational language ("observe," "notice," "report," "document")
- Absence of controlled trials or systematic reviews
- Small sample sizes ($N < 30$)

Implementation:

```
def detect_level_3(claim, sources):
    observational_keywords = ['observe', 'noticed', 'report', 'case', 'practitioner', 'traditional']
    has_observational = any(keyword in claim.lower() for keyword in observational_keywords)
    has_case_studies = any(s.type in ['case_report', 'case_series'] for s in sources)
    has_traditional = any('traditional' in claim.lower() or 'indigenous' in claim.lower())
    controlled_studies = any(s.type in ['randomized_trial', 'controlled_trial'] for s in sources)
    if (has_observational or has_case_studies or has_traditional) and not controlled_studies:
        return True, "Observation/Anecdote - Limited empirical data"
    return False, None
```

B.4 Level 4 - Correlation/Association

Definition: Statistical relationships identified through observational studies, surveys, or epidemiological research without established causal mechanisms. These show that variables are related but do not prove that one causes the other.

Characteristics:
- Statistical significance demonstrated
- Observational or epidemiological study designs
- No experimental manipulation
- Correlation coefficients or odds ratios reported
- Confounding variables may not be fully controlled
- "Association" or "correlation" explicitly stated

Examples:
Turmeric Domain:
- "Research shows a correlation between turmeric intake and lower inflammation markers."
- "Epidemiological studies associate higher curcumin consumption with reduced arthritis prevalence."
- "Population surveys indicate that regions with high turmeric use have lower rates of certain cancers."

Other Domains:
- "Studies find an association between coffee consumption and reduced Parkinson's risk." (correlation, not causation)
- "Higher education levels correlate with increased life expectancy."
- "Ice cream sales are correlated with drowning deaths." (spurious correlation)

Detection Criteria:
- Use of correlation/association terminology
- References to observational or cohort studies
- Statistical measures reported (r, OR, RR, p-values)
- Absence of experimental manipulation or causal language
- May include cautions about correlation vs. causation

Implementation:
```
def detect_level_4(claim, sources):
    correlation_keywords = ['correlation', 'association', 'linked', 'related to', 'associated with']
    has_correlation = any(keyword in claim.lower() for keyword in correlation_keywords)
    has_observational = any(s.type in ['cohort_study', 'cross_sectional', 'survey'] for s in sources)
    has_statistics = re.search(r'(r\s*=|p\s*[<>=]|OR\s*=|RR\s*=)', claim)
    causal_language = any(word in claim.lower() for word in ['causes', 'leads to', 'results in'])
    experimental = any(s.type in ['randomized_trial', 'controlled_experiment'] for s in sources)

    if (has_correlation or has_observational or has_statistics) and not (causal_language and experimental):
        return True, "Correlation/Association - Statistical link without causation"
    return False, None
```

B.5 Level 5 - Causal Explanation/Theory

Definition: Well-supported causal claims based on controlled experiments, mechanistic studies, or validated theoretical frameworks. These demonstrate not just that variables are related, but how and why one influences the other.

Characteristics:
- Experimental manipulation of variables
- Randomized controlled trials (RCTs) or laboratory experiments

- Causal mechanisms identified and validated
- Multiple studies confirming cause-effect relationship
- Theoretical models with predictive power
- Peer-reviewed publication in reputable journals

Examples:
Turmeric Domain:
- "Curcumin in turmeric inhibits inflammatory pathways by blocking NF-κB activation, as shown in controlled cell culture experiments."
- "Randomized trials demonstrate that curcumin supplementation reduces joint pain in arthritis patients."
- "Laboratory experiments confirm that curcumin induces apoptosis in cancer cells through mitochondrial pathways."

Other Domains:

- "Antibiotics kill bacteria by disrupting cell wall synthesis, as demonstrated in controlled laboratory studies."
- "Smoking causes lung cancer through DNA damage from carcinogens, confirmed by experimental models."
- "Gravitational forces cause planetary orbits, as predicted by general relativity and confirmed by observations."

Detection Criteria:
- Explicit causal language ("causes," "leads to," "results in," "induces")
- References to RCTs, controlled experiments, or mechanistic studies
- Description of causal pathways or mechanisms
- Use of technical terminology indicating deep understanding
- Multiple independent studies confirming the relationship

Implementation:
```
def detect_level_5(claim, sources):
    causal_keywords = ['causes', 'leads to', 'results in', 'induces', 'inhibits', 'activates', 'mechanism']
    has_causal = any(keyword in claim.lower() for keyword in causal_keywords)
    has_rct = any(s.type in ['randomized_trial', 'controlled_trial'] for s in sources)
    has_mechanism = re.search(r'(pathway|mechanism|activat|inhibit|modulat|target)', claim.lower())
    num_supporting = len([s for s in sources if s.supports_claim])
    if has_causal and (has_rct or has_mechanism) and num_supporting >= 2:
        return True, "Causal Explanation - Experimental evidence of mechanism"
    return False, None
```

B.6 Level 6 - Validated Knowledge/Consensus

Definition: Widely accepted facts supported by extensive research, meta-analyses, systematic reviews, and broad expert consensus across the field. These represent the current state of established knowledge in a domain.

Characteristics:
- Multiple high-quality studies in agreement
- Systematic reviews or meta-analyses synthesizing evidence
- Recognized by professional organizations and experts
- Taught in textbooks and standard curricula
- Replicated across different populations and contexts
- Minimal scientific controversy

Examples:

Turmeric Domain:

- "Turmeric's anti-inflammatory effects are recognized by both pharmacologists and traditional medicine practitioners, supported by numerous clinical trials."
- "The USDA and WHO acknowledge curcumin as a safe dietary supplement with established antioxidant properties."
- "Medical textbooks include curcumin among evidence-based complementary treatments for inflammatory conditions."

Other Domains:

- "Vaccines prevent infectious diseases through immune system priming, as established by decades of research and public health data."
- "DNA carries genetic information via nucleotide sequences, a fundamental principle of molecular biology."
- "Climate change is primarily driven by anthropogenic greenhouse gas emissions, as confirmed by 97% of climate scientists."

Detection Criteria:

- References to systematic reviews, meta-analyses, or clinical guidelines
- Mention of professional organization consensus (WHO, NIH, medical societies)
- Multiple citations from different research groups
- Absence of significant scientific controversy
- Terminology indicating consensus ("established," "recognized," "accepted")

Implementation:

```
def detect_level_6(claim, sources):
    consensus_keywords = ['recognized', 'established', 'accepted', 'consensus', 'standard', 'guideline']
    has_consensus = any(keyword in claim.lower() for keyword in consensus_keywords)

    has_metaanalysis = any(s.type in ['meta_analysis', 'systematic_review'] for s in sources)
    has_guidelines = any(s.source_type == 'clinical_guideline' for s in sources)
    num_sources = len(sources)
    agreement_rate = len([s for s in sources if s.supports_claim]) / num_sources if num_sources > 0 else 0
    if (has_consensus or has_metaanalysis or has_guidelines) and num_sources >= 5 and agreement_rate > 0.8:
        return True, "Validated Knowledge - Expert consensus with extensive evidence"
    return False, None
```

B.7 Level 7 - Demonstrated Fact/Proof

Definition: Definitive, reproducible knowledge confirmed through rigorous replication, often with mathematical precision or quantitative validation. These represent the highest level of certainty, approaching scientific laws or proven theorems.

Characteristics:

- Extensively replicated across independent labs and contexts
- Mathematical or quantitative formulation
- Predictive power with high precision
- Universal acceptance in the scientific community
- Often describes fundamental physical, chemical, or biological principles
- May include proven mathematical theorems or physical constants

Examples:

Turmeric Domain:

- "The molecular structure of curcumin ($C_{21}H_{20}O_6$) has been definitively characterized through X-ray crystallography and NMR spectroscopy."

- "Curcumin's absorption spectrum shows characteristic peaks at 420 nm, a reproducibly measured physical property."
- "The reduction in IL-6 levels following curcumin administration has been quantified across 50+ independent studies with consistent effect sizes ($d = 0.67$, 95% CI: 0.54-0.80)."

Other Domains:
- "Water boils at 100°C at standard atmospheric pressure (101.325 kPa), a reproducible physical constant."
- "The speed of light in vacuum is exactly 299,792,458 meters per second, as defined and measured."
- "The Pythagorean theorem ($a^2 + b^2 = c^2$) is a proven mathematical fact for right triangles."
- "DNA replication is semi-conservative, as definitively demonstrated by Meselson-Stahl experiment and replicated globally."

Detection Criteria:
- Quantitative precision with measurements and units
- Mathematical formulations or proven theorems
- Reference to physical constants or chemical structures
- Extensive replication (50+ independent confirmations)
- Universal acceptance without controversy
- Predictive models with >95% accuracy

Implementation:
```
def detect_level_7(claim, sources):
    precision_indicators = re.search(r'(\d+\.?\d*\s*(nm|m/s|°C|kPa|mol|Da|Å))', claim)
    mathematical_proof = re.search(r'(theorem|proof|equation|formula|constant)', claim.lower())
    chemical_structure = re.search(r'(C\d+H\d+O\d+|molecular formula)', claim)
    num_replications = len([s for s in sources if 'replication' in s.keywords])
    agreement_rate = len([s for s in sources if s.supports_claim]) / len(sources) if sources else 0
    is_quantitative = precision_indicators or mathematical_proof or chemical_structure
    is_extensively_replicated = num_replications >= 10 or agreement_rate > 0.95
    if is_quantitative and is_extensively_replicated:
        return True, "Demonstrated Fact - Reproducible quantitative knowledge"
    return False, None
```

B.8 Implementation Summary: Epistemic Level Decision Tree

For each claim:

```
├── No sources & subjective language? → Level 1 (Opinion)
├── Preliminary/pilot studies only? → Level 2 (Hypothesis)
├── Case reports/anecdotes/traditional knowledge? → Level 3 (Observation)
├── Correlation shown but no mechanism? → Level 4 (Association)
├── RCTs/experiments showing causation? → Level 5 (Causal Theory)
├── Meta-analyses/consensus/guidelines? → Level 6 (Validated)
└── Quantitative/replicated (50+)/mathematical? → Level 7 (Demonstrated Fact)
```

B.9 Epistemic Hierarchy Reference Table

Level	Category	Evidence Type	Study Design	Example Indicators	Confidence
1	Opinion/Belief	None	N/A	"might," "I think," no sources	Very Low
2	Speculation/Hypothesis	Preliminary	Pilot studies	"suggest," "may," "potential"	Low
3	Observation/Anecdote	Limited empirical	Case reports, traditional knowledge	"practitioners observe," case studies	Moderate-Low
4	Correlation/Association	Statistical	Cohort, cross-sectional	"correlated with," "associated," epidemiology	Moderate
5	Causal Explanation	Experimental	RCTs, controlled experiments	"causes," "mechanism," experimental validation	Moderate-High
6	Validated Knowledge	Systematic synthesis	Meta-analyses, guidelines	"consensus," "established," systematic reviews	High
7	Demonstrated Fact	Reproducible proof	Extensive replication, mathematics	Quantitative precision, universal acceptance	Very High

B.10 Practical Application Guidelines
For Medical/Health Claims:

- Require minimum Level 5 for treatment recommendations
- Level 6-7 for standard-of-care interventions
- Clearly distinguish traditional knowledge (Level 3) from experimental validation (Level 5+)

For Scientific Facts:

- Fundamental principles should be Level 6-7
- Emerging theories may be Level 4-5
- Hypotheses under investigation are Level 2-3

For Cultural/Traditional Knowledge:

- Long-standing practices may be Level 3 (observation-based)
- When validated by modern research, elevate to Level 5-6
- Respect epistemic pluralism: different knowledge systems may rate differently on this scale

User Interface Recommendations:

- Color coding: Red (1-2), Yellow (3-4), Green (5-6), Blue (7)
- Tooltips explaining evidence level
- Links to supporting sources
- Option to filter by minimum evidence level

End of Appendices