From Data to Evidence: AI-Enabled, Risk-Informed Decision Support for Nuclear Licensing

D. Mandelli [a*], C. Wang [a], K. O'Rear[a]
[a] Idaho National Laboratory, 2525 N. Fremont Ave, 83415 Idaho Falls (ID), USA

Abstract: Drafting safety analysis reports for nuclear reactors under regulatory guidance such as NUREG-1537 requires strict traceability, consistency across heterogeneous document sets, and an auditable use of evidence. Conventional approaches based on large language models (LLMs) do not reliably satisfy these requirements; they may produce text unsupported by the applicant's own documentation, introducing compliance risks in safety-critical contexts. We propose a retrieval-first, graph-augmented architecture in which the evidence chain (not the LLM) is the authoritative source. The framework couples a Chroma vector database for fine-grained semantic retrieval with a knowledge graph capturing the document topology and model-based systems engineering architecture of the system under consideration. An orchestration layer blends these two data structures into a structured *ContextPack* that constrains LLM generation to content traceable to specific source documents. The role of the LLM is to synthesize and articulate evidence, not to supply it. A set of guardrails enforces citation traceability, surfaces information gaps, and flags sentences that cannot be grounded in the retrieved evidence (making uncertainty explicit and auditable rather than eliminating it). A pilot study based on the ISU AGN-201M research reactor final safety analysis report quantitatively demonstrates the approach, in which 53 NUREG-1537 sections are drafted based on a corpus of documents that have been collected throughout the years of operating the Idaho State University AGN-201M.

Keywords: nuclear licensing, retrieval-augmented generation, knowledge graph, MBSE

1. INTRODUCTION

Nuclear reactor licensing is a document-based and highly labor-intensive process. In the United States, nuclear test reactor applicants are required to submit a final safety analysis report (FSAR) organized according to NUREG-1537 [1]. This Nuclear Regulatory Guide (NUREG) document provides details about FSAR format and content requirements (from reactor design description and thermal-hydraulic analysis to operational procedures and emergency planning). Each FSAR chapter must demonstrate compliance with other regulatory requirements, cite applicable standards (American National Standards Institute and American Nuclear Society [ANSI/ANS], Institute of Electrical and Electronics Engineers, American Society of Mechanical Engineers, and Nuclear Regulatory Commission [NRC] regulatory guides), and remain internally consistent across sections written by different authors over months or years.

These requirements impose three constraints that are difficult to satisfy with standard artificial intelligence (AI) based writing tools [2]. The first constraint is *traceability*: every factual claim must be tied to a specific source document and location so that NRC reviewers can verify provenance. The second is *cross-section consistency*: a parameter (such as rated thermal power) must be stated identically throughout the FSAR, and any discrepancy triggers a request for additional information (RAI), which adds additional burdens to the applicants and delays license approval. The third is *completeness*: each NUREG-1537 section specifies precisely required topics, and omitting any of them delays the review

* Corresponding author, diego.mandelli@inl.gov

cycle.

Large language models (LLMs) [3] offer speed advantages for drafting structured technical documents, but their use in safety-critical regulatory documentation introduces nontrivial risks. Without grounding in the applicant's actual source documents, LLMs may generate plausible sounding but factually incorrect content, cite standards or parameter values not found in the evidence base, or silently omit required topics. In a licensing context, an unsupported claim that reaches the submitted FSAR constitutes a compliance deficiency that can delay license approval or require a costly document revision.

The goal is not simply to reduce the incidence of such errors, it is to make them *visible* before submission. We propose that, in a regulatory setting, a framework that constrains evidence pathways and surfaces uncertainty is more defensible than one that claims to eliminate errors outright. In this respect, our main goal has been developing an auditable drafting process in which every generated sentence is either tied to a specific source document or explicitly flagged as unresolved.

This paper presents a framework where the principle guiding every output is that the "evidence chain is authoritative, not the LLM". The role of the LLM is to synthesize and articulate evidence retrieved from the applicant's own documents; it does not supply facts from its training data. A knowledge graph [4] captures the relationships among retrieved evidence, and a layered guardrail system makes gaps and uncertain inferences explicit rather than hiding them in fluent-sounding prose.

Here, the framework is demonstrated on a current licensing activity: the preparation of an FSAR for the Idaho State University (ISU) AGN-201M research reactor.

2. FRAMEWORK OVERVIEW

The developed framework is a Graph-Augmented Retrieval-Generation (Graph-RAG) system that extends standard retrieval-augmented generation [5, 6] by coupling a semantic vector search with a structured knowledge graph. Where conventional RAG retrieves text chunks by embedding similarity alone, Graph-RAG additionally traverses document topology and model-based systems engineering (MBSE) component relationships, providing the LLM with relational context that embedding similarity cannot express. This architectural choice is central to meeting the traceability, consistency, and completeness requirements of nuclear regulatory drafting.

The framework transforms a heterogeneous corpus of source documents into traceable regulatory draft text through five sequential stages, illustrated in Figure 1: parsing and chunking, entity annotation, knowledge store population, retrieval and context assembly, and controlled generation with guardrails. These stages are described in the following sections.

A central design principle is the evidence contract: a compact metadata record attached to every piece of information (i.e., a chunk) from the moment it is extracted from a source document through to the generated draft. This record carries the document name, source location (page number, section title), an ingestion timestamp, and a content hash. Because this record travels unchanged through every stage, any sentence in the final draft can be traced back to the exact paragraph, table row, or figure caption that supported it—providing the auditable evidence trail required by NRC reviewers.

3. DOCUMENT INGESTION

3.1. Parsing Heterogeneous Document Formats

The considered AGN-201M document repository includes a large amount of files with different file formats. As an example, it includes original design documents in PDF (some scanned), licensing reports and amendment letters in DOCX, equipment tables in XLSX, and internal notes in plain text. A modular document parser routine has been developed to route each file to the appropriate reader based

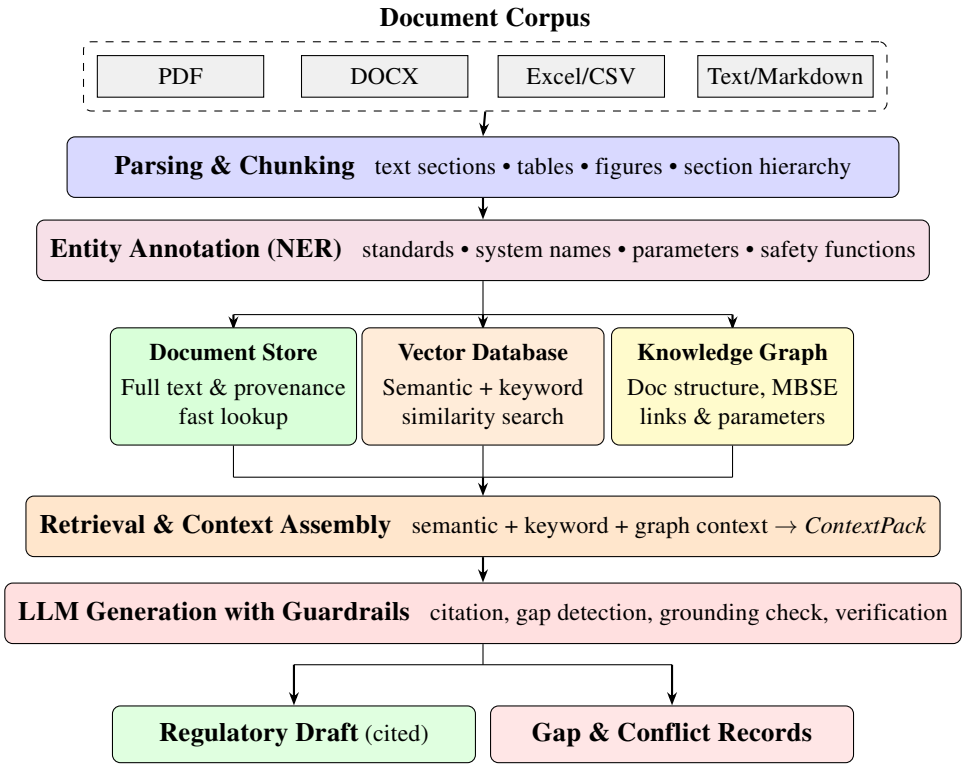

Figure 1: End-to-end pipeline from document corpus to regulatory draft. Source documents are parsed into text chunks, tables, and figures and then annotated with nuclear-domain entities. Three complementary knowledge stores are populated from the annotated chunks. A retrieval orchestrator fuses semantic search, keyword matching, and graph context into a single *ContextPack* that constrains LLM generation. Guardrails enforce citation traceability, surface missing topics, and flag potentially unsupported claims.

on file extension and extract its content in a normalized form.

Each document is divided into *chunks*: self-contained units of information that can be retrieved and cited independently. Note that here chunking is performed on a heading-by-heading basis (rather than creating fixed-length chunks). This is intentional so that a paragraph discussing, for example, coolant flow stays together rather than being split midsentence. Three types of chunks are produced:

- **Text chunks**: narrative paragraphs and lists are extracted with their section title and heading level preserved.
- **Table chunks**: each table is extracted as a structured record, preserving column headers, row data, caption, and the page number on which it appears. For retrieval, the table is presented to the LLM as a compact, pipe-delimited text rendering that conveys the same information without losing column alignment.
- **Figure chunks**: figure captions are extracted, and if a vision model is available, the figure image itself is described in natural language. The resulting text description is stored alongside the caption, enabling a semantic search over figures even when their content is graphical rather than textual.

Every chunk is marked with a precise evidence contract: document name, source path, page number, section title, ingestion timestamp, and a content hash. Note that this record travels unchanged through every downstream stage, ensuring that any claim in the final draft can be traced to its exact source.

3.2. Entity Annotation

Before a chunk is indexed, a two-tier named entity recognition (NER) pipeline annotates it with domain-specific entities [7]. The Tier 1 NER uses deterministic pattern matching covering eight categories: regulatory standards references (Title 10 of the Code of Federal Regulations [10 CFR], NUREG, ANSI/ANS, Institute of Electrical and Electronics Engineers, API, American Society of Mechanical Engineers, IEC, IAEA), plant system acronyms, accident and transient types (LOCA, LOFA, RIA), physical parameters with units, requirement identifiers (e.g., GDC 10), internal document cross-references, safety classifications, and safety function assignments. Each matched entity is normalized to a canonical identifier (e.g., all text variations of 10 CFR 50.46 map to the same reference key) so that cross-document comparisons are reliable.

The Tier 2 NER applies a statistical natural language processing (NLP) model [8] with nuclear-domain vocabulary patterns for nuclear-related entities (e.g., degradation modes, failure mechanisms, components, and equipment).

3.3. Amendment-Aware Document Registry

Since the beginning of development, it was clear that the document repository evolves over time in a licensing context; in particular, new versions of existing documents might be added. This poses challenges from an evidence retrieval point of view. To tackle this issue, the framework tracks each document as an instance of a *logical* identifier (e.g., the technical specifications document) with a specific amendment number and effective date. When multiple versions of the same document are present in the corpus, the registry assigns a lifecycle status: *active* (most recent), *superseded* (an older version), or *reference* (external guidance such as a regulatory guide). For retrieval, active document chunks are prioritized; superseded chunks are retrieved only when the active corpus cannot provide sufficient evidence, preventing silent evidence gaps caused by document turnover.

4. KNOWLEDGE STORE POPULATION

After parsing and annotating, each chunk is indexed into three complementary stores. The stores are not redundant (each supports a different kind of question), and the retrieval orchestrator queries all three in parallel when assembling evidence for a draft section.

4.1. Document Store: Full-Text Index

The Document Store holds the complete text of every chunk along with its evidence contract metadata. It functions as a fast lookup table; given a chunk identifier, the store returns the full text, table rows, or figure caption in milliseconds. The vector database (Section 4.2) stores compact embeddings for a similarity search but not full text; the Document Store fills that gap, providing the complete evidence snippet that the LLM needs to generate a grounded sentence.

4.2. Vector Database: Semantic and Keyword Search

Each text chunk—including table and figure descriptions—is converted into a numerical vector (i.e., an embedding) that captures its semantic meaning. These vectors are stored in a Chroma[1] vector database [5]. When a query arrives, the database finds the chunks whose embeddings are most similar to the query embedding, surfacing conceptually related content even when the exact words differ (e.g., a query about "primary coolant pressure" retrieves chunks describing "reactor coolant system operating conditions" without a literal keyword match).

Alongside the embedding index, the framework maintains a BM25 keyword index [9] over the same chunks. BM25 is effective for queries that contain specific regulatory identifiers (e.g., 10 CFR 50.46) or parameter names, where exact wording matters. Both indices are populated from the same annotated chunk records and kept in sync.

NER annotations are stored as metadata alongside each chunk in the vector database. This enables query-time filtering. If a query asks about a LOCA transient, the retriever can restrict results to chunks that mention LOCA, preventing general coolant system guidance from large reactor documents from appearing in results for a small research reactor query.

4.3. Knowledge Graph: Document Structure and MBSE Relations

The knowledge graph (implemented in the Neo4j[2] [10]) captures relationships between pieces of information that a vector similarity score cannot express. It is populated from the same annotated chunks as the other stores but stores connections rather than text.

The graph contains these seven node categories:

- *Document* nodes represent individual ingested files
- *Section* nodes represent logical document sections linked to their parent document in linear order, so the graph encodes where each section falls in the overall document structure
- *Text chunk*, *table*, and *figure* nodes represent individual content units, linked to their parent sections, and carry indexing flags and content hashes but not full text, keeping the graph compact
- *Standard* nodes represent regulatory references such as `10 CFR 50.46` (one node per unique reference, shared by all chunks that cite it)
- *MBSE component* nodes represent plant system elements (a control rod assembly, a coolant pump, a scram circuit) referenced in the documents, optionally linked to an external MBSE system model.

Additional NER-derived nodes cover physical parameters (value, unit, and parameter type), transient event types, regulatory requirements, and safety functions.

The graph encodes two kinds of relationships:

[1] Chroma vector database; `https://docs.trychroma.com`.
[2] Neo4j graph database; `https://neo4j.com`.

- *Structural* edges represent document topology: a chunk belongs to a section, a section belongs to a document, and sections are ordered sequentially. The sequential-order edges are particularly useful because they allow the retrieval orchestrator to fetch not just a matched chunk but also the sections immediately before and after it, providing context that a similarity score alone cannot provide.
- *Semantic* edges, derived from NER annotations, connect chunks to the standards they reference, the MBSE components they describe, the transients they address, the regulations they cite, and the physical parameters they report.

The graph construction requires a two-pass scan of the chunk records. The first pass only processes text chunks to build a section-order map, because tables and figures reference their parent section by title rather than by position and that title must be resolved to a section index before edges can be created. The second pass processes all chunk types, emitting nodes and edges in document order. Nodes are inserted with an *upsert* strategy, so re-running ingestion after adding new documents does not create duplicates.

For each chunk returned by the vector search, the orchestrator issues a graph query that retrieves the chunk's parent section and document, the sections adjacent to it, all regulatory standards it references, and all MBSE components it mentions. This structured neighborhood is appended to the ContextPack alongside the chunk text, giving the LLM access to relational evidence that an embedding similarity cannot surface—for example, that two chunks from different documents both cite the same safety requirement or that a parameter value in Chapter 4 is also stated in Chapter 7.

5. HYBRID RETRIEVAL AND CONTEXT ASSEMBLY

5.1. Fusing Semantic and Keyword Results

For each section of the draft, the retrieval orchestrator issues both a semantic (vector similarity) query and a keyword (BM25) query, then merges the two ranked result lists using reciprocal rank fusion (RRF) [11]. The fused score for a chunk c is:

$$s_{\mathrm{RRF}}(c) = \sum_{v} \frac{w_v}{k_{\mathrm{RRF}} + r_v(c) + 1}, \tag{1}$$

where v indexes the retrieval method, $r_v(c)$ is the rank of chunk c in method v, w_v is a method weight (1.0 for semantic, 1.5 for keyword), and $k_{\mathrm{RRF}} = 60$ is a smoothing constant. Note that semantic and keyword scores operate on incompatible numerical scales. RRF avoids the need to normalize them by operating on ranks rather than raw scores, making the fusion robust across different types of queries.

5.2. Evidence Sufficiency Assessment

Before the LLM is invoked, the orchestrator evaluates whether the retrieved chunks provide adequate coverage for each topic that the NUREG section requires. Coverage is scored by comparing retrieved chunks against each required topic using semantic similarity, with type-specific thresholds (Table 1). Topics for which no retrieved chunk meets the minimum threshold are flagged as *zero-evidence* topics. These flags are shown to the analyst before generation begins, allowing them to add missing source documents or adjust the retrieval query without spending LLM processing time on a section that lacks adequate evidence.

5.3. ContextPack Assembly

The top-ranked chunks from RRF fusion (together with their graph neighborhoods from the knowledge graph expansion) are assembled into a single structured *ContextPack*. This package—containing chunk texts, provenance metadata, graph facts, and the sufficiency assessment—is the sole input for the LLM generation stage. The LLM never queries the knowledge stores directly; all evidence it sees has already passed through the retrieval and sufficiency checks.

Table 1: Pre-draft evidence sufficiency thresholds by topic type.

Topic type	Covered	Near-miss
System description	≥ 0.40	0.30–0.40
Parameter value	≥ 0.35	0.25–0.35
Analysis result	≥ 0.20	0.10–0.20
Design basis	≥ 0.30	0.20–0.30

6. CONTROLLED GENERATION WITH GUARDRAILS

6.1. Section Directive Templates

Each NUREG-1537 section is governed by a set of section directives: a structured template that specifies which required topics the section must address (with their regulatory basis), a retrieval hint to steer evidence gathering toward reactor-specific evidence rather than generic guidance, and an output format specification. The directive also records which systems are not applicable to the specific reactor design. This last point is critical for research reactor licensing, as the AGN-201M lacks systems that full-power reactor guidance discusses at length, such as emergency core cooling, secondary coolant loops, and cover gas management. Without an explicit not-applicable declaration, the LLM may generate plausible descriptions of these nonexistent systems by drawing on generic NRC guidance documents. When the directive marks a system as not applicable, generation for that section is short-circuited and replaced with a brief not-applicable statement, preventing phantom system hallucinations.

6.2. Guardrail Stack

After the *ContextPack* is assembled, text generation proceeds through four guardrail layers:

1. **Citation-Constrained Prompting.** The LLM is instructed to cite every factual claim by referencing the specific chunk in the *ContextPack* from which it was drawn. The generation output is kept in two forms: a fully cited version (the primary audit record) and a clean version with citations removed for readability. Any NRC reviewer can inspect the cited version to verify that every sentence in the draft has a corresponding source.

2. **Gap Marker Extraction.** The LLM is instructed to insert an explicit marker (*[INFORMATION NOT AVAILABLE: topic]*) wherever a required topic cannot be addressed from the provided evidence. After generation, these markers are collected, deduplicated, and merged with the pre-draft sufficiency flags into a complete gap record for the section. The gap record tells the licensing team exactly which topics the draft covers and which ones require additional source documents—a direct input for an RAI risk assessment.

3. **Semantic Grounding Check.** Each sentence in the generated draft is compared to all chunks in the *ContextPack* using the same embedding model used for retrieval. Sentences whose highest cosine similarity to any evidence chunk falls below a threshold (0.30) are flagged as potentially ungrounded and are surfaced for human review. This check does not automatically reject flagged sentences—introductory framing sentences often have no direct evidence counterpart—but it converts the uncertainty from invisible to explicit, which is the appropriate goal in a regulatory drafting context.

4. **Verification Pass (Optional).** A second LLM instance, configured as an independent judge, is optionally invoked on the flagged sentences. The judge assesses whether each sentence contradicts or lacks support from the evidence chunks. Its assessments are recorded in a structured audit log that can be submitted alongside the SAR to demonstrate that the AI-assisted drafting

process was subjected to independent review.

6.3. Cross-Section Parameter Conflict Detection

After all sections are drafted, the framework scans the full draft set for physical parameter mentions—values with units such as power in watts, temperature in degrees Celsius, or pressure in psi. Parameters that appear in multiple sections with different values are flagged as conflicts and are surfaced for human resolution before submission (e.g., rated thermal power stated as 5 W in the reactor description chapter and 5.5 W in the thermal analysis chapter). This step directly targets the cross-section consistency requirement that generates the most NRC RAIs in research reactor licensing applications.

7. ISU AGN-201M CASE STUDY

7.1. Document Corpus and Licensing Context

The corpus assembled for this work includes approximately 140 documents organized in three tiers.

- Tier 1 (primary evidence) includes the 2024 Safety Analysis Report amendment, license conditions, technical specifications, and license amendment documents.
- Tier 2 (supporting evidence) includes historical SARs from 2003 and 2021, fuel characterization reports, and operational procedure manuals.
- Tier 3 (guidance) includes NUREG-1537, applicable NRC regulatory guides, and ANSI/ANS standards. The amendment registry marks Tier 1 documents as active, Tier 2 versions as superseded, and Tier 3 guidance as reference, ensuring that retrieval prioritizes the most current evidence.

7.2. Ingestion and Knowledge Store Population

The Document Store resulting from the ingestion of the Tier 1 and Tier 2 corpus contained approximately 3,800 chunk records spanning text chunks, tables, and figures. Of these, roughly 2,600 text and table chunks were embedded into the vector database. The knowledge graph was populated with approximately 1,200 content nodes (text chunks, tables, figures), 340 section nodes, 28 document nodes, and 890 NER-derived nodes representing standards references, MBSE component usages, and physical parameter instances—approximately 2,460 nodes in total.

One challenge specific to scanned documents was that some tables had their section header on a preceding page. The two-pass graph build resolved all such table-to-section assignments correctly; a single-pass approach would have misassigned those tables to the wrong section.

7.3. Pilot Results: Quantitative Summary

Table 2 summarizes the quantitative outcome of the two-phase drafting run.

7.4. Observations by Failure Mode

All 53 NUREG-1537 sections were drafted in a single overnight run. Six sections covering systems absent from the AGN-201M (e.g., emergency core cooling, secondary coolant, cover gas) were handled by not-applicable declarations and did not require LLM generation. A post-run review identified three categories of failure among the 47 LLM-drafted sections. Three sections produced repetitive output (one exceeding 180,000 characters), which was caused by the LLM looping on gap markers when the maximum output length was not bound. One section on the reactor control system was aborted because the assembled evidence package exceeded the model's context window (30,499 tokens against a 32,768-token limit). Two sections contained factual errors: one described the reactor fuel as highly enriched uranium when the AGN-201M uses low-enriched uranium and one omitted the fuel's distinctive UO_2-in-polyethylene composition entirely. In both cases, the retriever had surfaced generic

Table 2: ISU AGN-201M pilot: quantitative summary across both phases.

Ingestion	
Documents processed	≈140
Chunk records	≈3,800
Vector-indexed chunks	≈2,600
Knowledge graph nodes	≈2,460
Wall time	≈20 min
Phase 2 — Re-draft	
Sections re-drafted	9/47 (19%)
Resolved via directive tuning	9/9 (100%)
Code changes required	0
Parameter conflicts detected	3
Unsupported claims remaining	0

Phase 1 — Full Draft	
Total NUREG-1537 sections	53
Handled by N/A declaration	6 (11%)
LLM-drafted sections	47 (89%)
Sections with gap markers	28/47 (60%)
Gap markers in draft	201
N/A-system unsupported claims	6
Critical factual errors	4
Generation time / section	≈75 sec
Total generation wall time	≈90 min

NRC guidance written for different reactor types, and the directive lacked sufficient specificity to steer retrieval away from that guidance.

In all six N/A-system cases, the not-applicable declaration mechanism did not prevent the generation of wrong information; these were sections that had not yet been declared N/A during Phase 1 setup. Once the declarations were added, the issue was resolved without LLM invocation. This distinction is important; the failure mode was due to an incomplete directive configuration, not an architectural deficiency.

Nine sections were re-drafted following directive tuning. All failures were resolved without any changes to the framework code, confirming that the architecture was sound and the failures were engineering configuration issues. The output length cap (4,096 tokens) was applied uniformly, eliminating the repetition loops. The context window overrun was resolved by reducing the number of retrieved chunks for that section from 16 to 4. The two fuel-description errors were resolved by updating the retrieval hint to explicitly anchor on AGN-201M LEU specifications, steering evidence gathering away from inapplicable guidance.

The parameter conflict scanner detected three cross-section inconsistencies: rated thermal power differed between Chapters 4 and 13 (one source citing a pre-amendment design limit), coolant flow rate appeared in two sections under different measurement conditions, and core excess reactivity was cited inconsistently across Chapters 4 and 14.

The 201 explicit gap markers in the Phase 1 draft represent an important outcome in their own right. In a conventional manual drafting process, these gaps would be invisible—an author might write around missing information, producing fluent prose that obscures the absence of evidence. In this framework, gaps are surfaced as explicit, countable records that can be directly mapped to specific required topics and assigned to the team members best positioned to resolve them. The 60% gap rate in Phase 1 reflects the incompleteness of the Tier 1 corpus, not a framework failure; it directed the document-gathering effort toward exactly the sections that needed it.

8. DISCUSSION

The governing design principle in this framework is the strict separation of evidence supply (from the knowledge stores) from evidence articulation (through LLMs). The knowledge stores (i.e., Document Store, vector database, and knowledge graph) are the authoritative sources of facts about the considered reactor. The LLM is a synthesis and writing tool that operates only on evidence already retrieved from those stores. This separation has a direct implication for licensing reviewers: auditing the draft means

auditing the evidence chain, not the LLM. A reviewer who questions a sentence in the draft can inspect the cited chunk directly and trace it to the source document, page, and section. The LLM's reasoning is not auditable and is not presented as authoritative.

It is important to note that the developed guardrails do not prevent the LLM from generating unsupported text (no system can make that guarantee at generation time). What the guardrails do is make uncertainty explicit and countable: gap markers record what the evidence base cannot support, semantic grounding flags sentences that lack a strong evidence match, and the verification pass produces a human-reviewable audit trail. We propose that a reasonable goal for AI-assisted regulatory drafting is not a system error-free drafting process but one that makes residual uncertainty visible so that human reviewers can focus their attention where it is most needed.

The Phase 1 pilot results illustrate this principle. The 201 explicit gap markers in the first-pass draft were not a failure but a signal—directing document-gathering effort to exactly the sections where evidence was weakest, before a human reviewer had to discover those gaps by reading 53 chapters of draft prose.

The framework reduces licensing risk along two axes:

- Commission risk (i.e., the risk of submitting a SAR that contains an unsupported claim) is bounded by citation-constrained prompting and semantic grounding; every generated sentence either cites a specific evidence chunk or is explicitly flagged for human review.
- Omission risk (i.e., the risk that a required NUREG topic is absent from the submitted section) is converted from a silent failure into an auditable gap record visible to the licensing team before the draft is finalized.

Moving the framework to a different reactor type requires updating the section directive templates (which required topics apply, which systems are not applicable) and adjusting the chapter structure if a different format guide applies. The embedding models, keyword index, and LLM backend are reactor-agnostic and require no retraining. The NER pattern library covers the major plant system acronyms and transient types across reactor classes. The principal adaptation cost is therefore a knowledge-engineering effort—updating templates and not-applicable declarations—not a machine-learning cost.

Lastly we would like to highlight two limitations that should be mentioned. The first is that the parameter conflict scanner compares values as text strings and does not normalize units, for example, "500 psi" and "3.45 MPa" would be flagged as a conflict even though they are equivalent (unit normalization will be explored in future work). The second is related to the semantic grounding threshold (0.30), which is a heuristic calibrated on the ISU corpus; for a reactor design whose source documentation uses highly specialized vocabulary not well represented in the embedding model, the threshold may need adjustment (or ideally, self-calibrated) to avoid excessive false-positive grounding flags.

9. CONCLUSIONS

This paper has presented a retrieval-first, graph-augmented framework for AI-assisted drafting of nuclear licensing documentation. The framework's central design principle is that the evidence chain—not the LLM—is authoritative; the LLM synthesizes and articulates evidence retrieved from the applicant's own documents, and the guardrail system makes uncertainty explicit and auditable rather than hiding it in fluent-sounding prose. This framing aligns with the core requirements of regulatory licensing: traceability, completeness verification, and auditable quality assurance.

The ISU AGN-201M pilot demonstrated the approach at operational scale; 53 NUREG-1537 sections

were drafted in approximately 90 minutes, with 201 explicit gap markers directing targeted document gathering, three cross-section parameter conflicts surfaced and resolved within one working day, and all Phase 1 failure modes corrected through directive configuration tuning without framework code changes. These results illustrate that the value of the framework lies not in producing error-free first drafts but in making errors and gaps visible—shifting reviewer efforts from discovery to resolution.

Future work will address unit normalization in the parameter conflict scanner, automated calibration of the grounding threshold for new reactor vocabularies, and integration with configuration management systems used in formal NRC regulatory review workflows.

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy, Office of Nuclear Energy. AI tools assisted with the development of the presented framework and the revision of this paper. All technical analyses, designs, results, and conclusions were developed and verified by the authors.

REFERENCES

[1] U.S. Nuclear Regulatory Commission, "Guidelines for preparing and reviewing applications for the licensing of non-power reactors: Format and content (nureg 1537, part 1)," Tech. Rep. NUREG-1537 Part 1, NRC, Washington, DC, 1996.

[2] F. Neha, D. Bhati, D. K. Shukla, A. Guercio, and B. Ward, "Exploring ai text generation, retrieval-augmented generation, and detection technologies: a comprehensive overview," 2024.

[3] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao, "Large language models: A survey," 2025.

[4] R. Liu, R. Fu, K. Xu, X. Shi, and X. Ren, "A review of knowledge graph-based reasoning technology in the operation of power systems," *Applied Sciences*, vol. 13, no. 7, 2023.

[5] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-T. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," *Advances in Neural Information Processing Systems*, vol. 793, pp. 9459–9474, 2020.

[6] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," 2024.

[7] C. Wang, D. Mandelli, and J. Cogliati, "Technical language processing of nuclear power plants equipment reliability data," *Energies*, vol. 17, no. 7, 2024.

[8] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-strength natural language processing in Python." `https://spacy.io`, 2020.

[9] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: Bm25 and beyond," vol. 4, pp. 1–174, 09 2009.

[10] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, and J. Larson, "From local to global: A graph rag approach to query-focused summarization," 2025.

[11] G. Cormack, C. Clarke, and S. Buettcher, "Reciprocal rank fusion outperforms Condorcet and individual rank learning methods," in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, (Boston, MA), pp. 758–759, 2009.