

A Role-Specialized Retrieval-Augmented Generation Framework for Automated Compliance Checking in Structural Engineering

Xiaoyan Lu¹, Jianwei Zhu^{1,*}

¹School of Construction Engineering, Guangxi Vocational & Technical Institute of Industry, Nanning, China

*Corresponding author: Jianwei Zhu (e-mail: sqe513@163.com).

ABSTRACT Automated code compliance checking in structural engineering remains difficult because practical systems must balance accuracy, maintainability, and deployment cost. Pure prompting with large language models is prone to hallucination and unstable numerical judgment, conventional retrieval-augmented generation may fail to assemble dispersed regulatory evidence, and ontology-heavy pipelines can be expensive to extend when codes evolve. We therefore propose a role-specialized retrieval-augmented generation framework composed of a deterministic Parser, a Reviewer guided by an Equilibrium Rule prompt, and a Critic that outputs a structured verdict. On a 211-case benchmark built from real-project building information modeling exports and expert review across slabs, beams, and columns, the framework achieves 96.68% overall accuracy, compared with 57.35% for zero-shot prompting, 79.15% for naive retrieval-augmented generation, 77.25% for naive retrieval-augmented generation with the Equilibrium Rule, and 89.10% for a re-implemented ontology- and rule-based baseline. The benchmark knowledge resources, retrieval corpus, and rule-supporting mappings are extracted from two Chinese national structural design codes covering concrete member design and structural loading. Ablation results suggest that the Equilibrium Rule and the Critic contribute most of the observed gain. We therefore position the framework as a decision-support approach for engineering compliance review rather than as a fully autonomous approval engine.

INDEX TERMS Automated code compliance checking, Building information modeling, Large language models, Retrieval-augmented generation, Structural engineering

I. INTRODUCTION

THE rapid digitalization of the Architecture, Engineering and Construction (AEC) industry has made Building Information Modeling (BIM) a central carrier of structural design information. BIM models can store member geometry, material grade, reinforcement information, load assumptions, and environmental conditions in a computable form, which creates a technical basis for automated structural review [1–2]. However, in many projects, checking whether slabs, beams, and columns satisfy national design codes still relies heavily on manual interpretation by experienced engineers, making the process time-consuming, difficult to reproduce, and vulnerable to omissions [3].

Existing automated code compliance checking (ACCC) systems have not yet fully solved this problem. Deterministic rule engines and ontology-based systems can be accurate within a predefined rule schema, but they require extensive manual formalization and continuous maintenance when codes are revised or when new component types are added [4–5]. Commercial BIM-checking platforms are

useful for geometric or clash-related checks, yet structural code compliance often depends on numerical thresholds, material combinations, seismic modifiers, and environmental exposure classes that are harder to encode in a general template.

Recent research has therefore explored natural language processing, large language models (LLMs), and retrieval-augmented generation (RAG) as more flexible alternatives [6]. These approaches reduce the need to translate every provision into symbolic logic, but pure prompting may hallucinate clauses, misread numerical limits, or ignore missing input fields. Conventional RAG improves grounding by retrieving regulatory evidence, but structural provisions are frequently dispersed across clauses, tables, and explanatory rules. Incomplete retrieval may consequently lead to unstable approvals, excessive NEED-MORE-INFO responses, or insufficiently justified rejections.

Against this background, this study examines whether a practical, role-specialized RAG pipeline can better balance accuracy, maintainability, and deployment cost. The pro-

posed framework separates deterministic query construction, grounded code review, and structured verdict extraction into three auditable stages: Parser, Reviewer, and Critic. Using a benchmark derived from real-project BIM exports and expert review under GB 50010-2010 and GB 50009-2012, the framework is compared with zero-shot prompting, naive RAG, an Equilibrium-Rule single-agent variant, and an ontology-SWRL baseline.

The contribution of this paper is therefore threefold. First, it constructs a 211-case benchmark whose parameter values come directly from in-service BIM models and whose verdict labels are reviewed by domain specialists. Second, it proposes a lightweight role-specialized RAG framework that keeps retrieval-query construction, clause-grounded reasoning, and verdict extraction auditable. Third, it reports overall, category-wise, component-wise, ablation, significance, and error-analysis results to clarify both the performance gains and the remaining failure modes. The goal is not to replace engineers, but to provide a traceable decision-support method for structural compliance checking.

II. RELATED WORK

A. RULE-BASED AND ONTOLOGY APPROACHES

Before the deep-learning era, ACCC research mainly relied on classical information extraction, semantic modeling, and manual logic mapping. Representative studies translated regulatory sentences into executable predicates or OWL/SWRL assertions, and then evaluated them with description-logic or rule-based reasoners [4,7–12]. These studies established the feasibility of machine-readable compliance checking, but they also showed that the text-to-formal-rule conversion process is costly, error-prone, and difficult to update across code editions.

Ontology-SWRL systems can be highly precise within their authored schema, but they incur well-documented maintenance costs. Every new material or design variable requires new OWL classes and inverse-role assertions, and the NLP-to-OWL routing step is itself stochastic. The literature consistently reports performance ceilings in the low-to-mid 80s on comparable code-compliance benchmarks [4][7][10][11], which the respective authors attribute to imperfect free-text-to-formal-syntax translation. Dimyadi and Amor [3], Nawari [13], and Z. Zhang, Ma, and Broyd [14] discuss rule interpretation and representation challenges for automated code compliance checking, alongside broader BIM-driven compliance practice, including commercial platforms such as Solibri and Navisworks. In contrast, the present study keeps the constraints in natural language and controls the workflow through prompt design, stage decomposition, and targeted knowledge patching. For comparison, we re-implement an ontology-SWRL baseline under the same LLM router, embedding model, and benchmark.

B. LLMS, RAG AND RETRIEVAL AUGMENTATION

The use of LLMs in AEC is a nascent but fast-growing area [15][16]. When tested on specific code-compliance

tasks (for example ACI-318 concrete design), however, zero-shot LLMs struggle to distinguish between similar numeric thresholds, presumably because specialized provisions are under-represented in their pre-training corpora [17]. Retrieval-Augmented Generation (RAG) was introduced by Lewis et al. [6] and extended by fusion-in-decoder [18] to anchor generative models to external evidence. A growing body of work targets the retrieval step itself—HyDE [19] and Query2Doc [20] expand queries with synthetic passages to raise recall on sparse domains, while Self-RAG [21] teaches a single model to decide when and what to retrieve. Survey-style work on trustworthy and grounded RAG [22] has highlighted the broader problem of retrieval incompleteness in knowledge-intensive tasks. Complementary construction-domain studies compare RAG against fine-tuned LLMs for safety-related knowledge retrieval [23]. Our pre-indexing corpus augmentation is positioned in the same lineage as HyDE/Query2Doc in spirit (injecting synthetic surface forms into the retrieval layer) but differs on three axes: (i) entries are authored manually by a domain expert rather than generated by an LLM, (ii) each entry encodes a verified quantitative threshold, and (iii) the entries are concatenated with the corpus before indexing rather than generated per query, which preserves deterministic retrieval and keeps the synthetic content auditable. Accordingly, this study treats retrieval completeness as a core design issue rather than a secondary implementation detail.

C. PROMPT DESIGN AND CHAIN-OF-THOUGHT

Several prompt-engineering techniques are relevant to the Equilibrium Rule. Chain-of-Thought prompting [24], Least-to-Most prompting [25], and Zero-shot Chain-of-Thought [26] show that lightweight prompt structure can materially change LLM behavior on reasoning-intensive tasks. In this paper, the Equilibrium Rule is used as a domain-specific prompt heuristic for structural review: when a factor is explicitly declared in the input but no applicable limit is retrieved for that factor, the Reviewer should avoid escalating to NEED-MORE-INFO unless a required parameter is actually missing. We therefore treat this rule as a practical design choice for the target task rather than as a general theory of LLM reasoning. This observation supports using the Equilibrium Rule as a targeted prompt constraint within the proposed Reviewer stage.

D. ROLE-SPECIALIZED LLM SYSTEMS

A third line of research decomposes complex reasoning into multiple LLM stages with different responsibilities. Frameworks such as AutoGen [27], MetaGPT [28], and AgentBench [29] suggest that role separation can improve robustness on long-horizon tasks, although implementations vary widely across applications. Our work adopts only a lightweight version of this idea: a deterministic parsing stage, a single reasoning stage, and a final verdict-extraction stage. The goal is not to claim a general multi-agent mechanism, but to examine whether this simple staged design

is useful for engineering code review. This line of work motivates the staged Parser-Reviewer-Critic organization used in the proposed framework.

III. METHODOLOGY

To address the practical limitations of current LLM prompting, conventional RAG, and ontology-based compliance checkers, we structure the compliance pipeline as a tripartite, role-specialized system in which each stage has a narrow and auditable role.

A. SYSTEM OVERVIEW

The proposed architecture, illustrated in Figure 1, maps a raw structural query onto regulatory knowledge resources extracted from GB 50010-2010 (Code for design of concrete structures) and GB 50009-2012 (Load code for the design of building structures). The pipeline consists of three collaborating modules that communicate via structured intermediate messages. The Parser is a deterministic, non-learning component: it applies a fixed dictionary that maps each BIM parameter field (e.g. thickness, concrete grade, and environment class) to canonical Chinese code terminology used in the retrieval corpus (e.g. slab thickness / concrete strength grade / environment class) and concatenates those tokens with the component type into a short retrieval query. This replaces the more common LLM-as-parser pattern and avoids introducing an additional generative step before retrieval. The Reviewer is a single LLM call that receives the Parser's query, the top-k retrieved chunks, and the Equilibrium Rule system prompt. The Critic is a second LLM call that consumes the Reviewer's free-text reasoning and returns a strictly typed JavaScript Object Notation (JSON) verdict, which gives it both a formatting and an auditing role (see Section 3.5).

Terminological note. We describe the system as a role-specialized pipeline: the three modules do not re-query each other in an open loop, and the Parser stage is deterministic rather than learned. The design focuses on separating retrieval-query construction (Parser), grounded reasoning (Reviewer), and verdict extraction (Critic).

B. RETRIEVAL DESIGN

As shown in Figure 1, the proposed framework follows a three-stage workflow. First, the Parser converts a BIM record into a short retrieval query using a fixed field dictionary rather than an LLM-generated reformulation. Second, the Reviewer receives the parsed query together with the retrieved regulatory passages and produces a natural-language assessment under the Equilibrium Rule prompt. Third, the Critic converts that assessment into a structured verdict: APPROVE, REJECT, or NEED-MORE-INFO.

Retrieval uses Facebook AI Similarity Search (FAISS) over a corpus built from text chunks and rule-supporting passages extracted from both GB 50010 and GB 50009, and optionally augmented with the handcrafted snippets described in Section 3.4. The retrieval setup is intentionally

lightweight: it improves the coverage of dispersed thresholds and table lookups without introducing a full symbolic knowledge base.

The augmentation entries are kept auditable and are used only to support retrieval coverage, not to introduce hidden decision rules.

C. EQUILIBRIUM RULE

The Equilibrium Rule is the main prompt rule used by the Reviewer. In plain terms, it says that when an environmental or seismic modifier is explicitly declared in the input but the retrieved text does not provide a corresponding limit, the Reviewer should not return NEED-MORE-INFO for that reason alone. NEED-MORE-INFO is reserved for cases in which a required input field is actually missing. This rule was introduced to reduce over-conservative behavior on compliant and boundary cases; its effect is evaluated empirically in Section 4 rather than argued as a standalone theoretical principle.

D. CORPUS AUGMENTATION

Before FAISS indexing, we concatenate 24 manually curated and keyword-rich snippets with the main regulatory corpus. Each snippet restates a verified quantitative limit in a retrieval-friendly form, such as a minimum slab thickness, a concrete-grade requirement, or an environmental-cover threshold. This operation is a retrieval-side aid rather than an additional reasoning agent; its purpose is to reduce missed retrieval of dispersed clauses and table-driven limits while keeping every added entry traceable and auditable.

E. CRITIC AGENT

The Critic receives the Reviewer's free-text assessment and converts it into a strictly structured verdict. It checks whether the reasoning supports APPROVE, REJECT, or NEED-MORE-INFO, verifies that the cited clauses and assumptions are reported in machine-readable fields, and prevents unsupported narrative statements from becoming final decisions. The Critic does not introduce new regulatory requirements; it functions as a formatting, consistency-checking, and audit layer before the decision-support output is returned to the engineer.

Figure 2 illustrates how sparse code evidence is complemented by handcrafted quantitative anchors before FAISS indexing, making the retrieval-side augmentation auditable and easy to inspect.

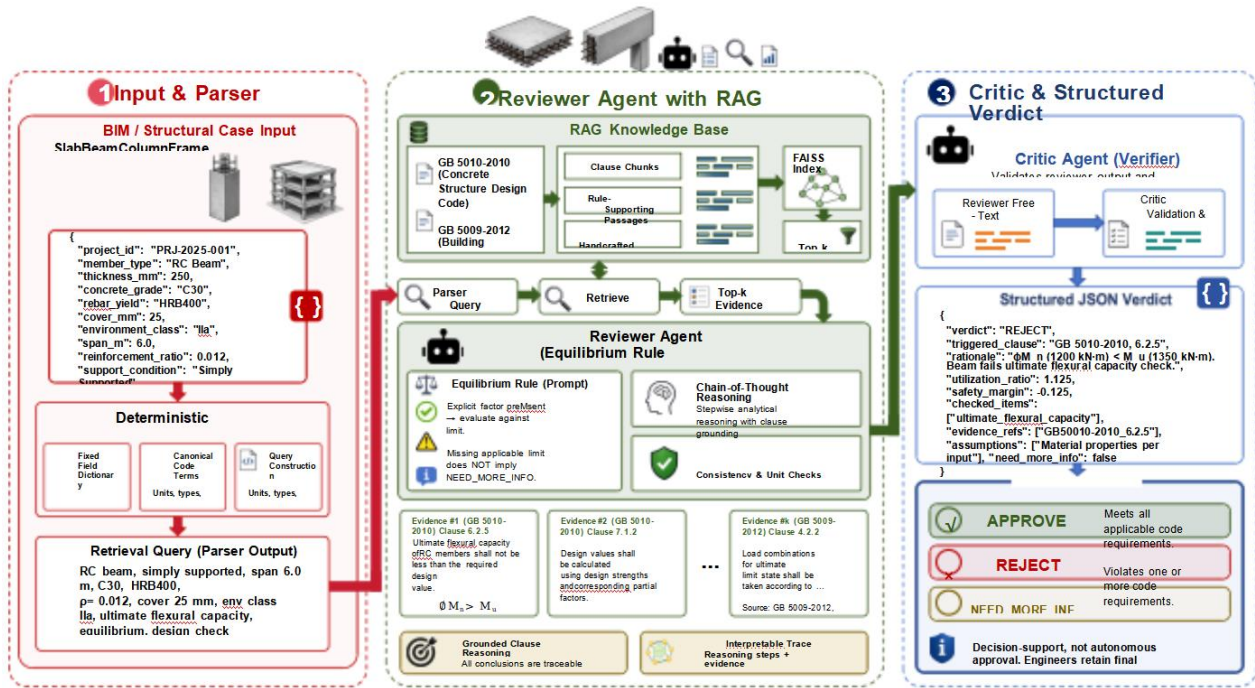


FIGURE 1. Role-specialized pipeline with Parser, Reviewer, and Critic modules.

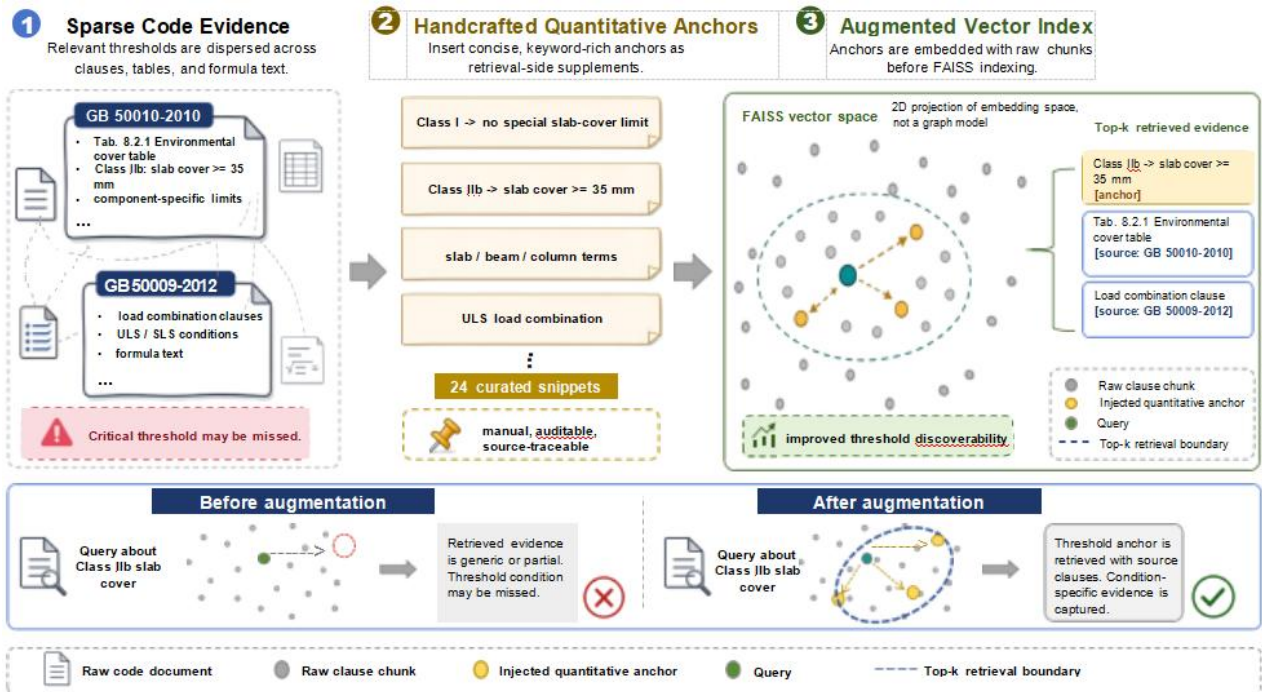


FIGURE 2. Retrieval-corpus augmentation before FAISS indexing.

IV. EXPERIMENTS AND RESULTS

A. DATASET, BASELINES AND IMPLEMENTATION

Because existing public datasets for Chinese structural review tasks are either too small or too narrowly scoped, we constructed a new benchmark, Structural Review Test Cases v3, from real-project BIM exports. The underlying param-

eter records are collected from BIM models of in-service structural-engineering projects and exported component by component through the native BIM platform (Revit/YJK) property dictionary. After export, each component record is converted into a structured JSON case by a deterministic field-mapping script that preserves the numerical values

verbatim. As summarized in Table I, the benchmark contains 211 cases distributed across three component types (slabs, beams, and columns) and five compliance categories.

The 211 cases are distilled from a raw pool of several thousand component records by stratified sampling over component type $\times$ five-category compliance status, so that each category is represented with sufficient statistical power without saturating any single project's component population. Category labels are determined through expert review against the applicable benchmark clauses extracted from GB 50010 and GB 50009, with a deterministic Python rule engine used only as an auxiliary consistency-checking tool during dataset construction. The final 211-case benchmark is drawn to approximate the empirically observed category frequencies while retaining enough boundary and incomplete-information cases to stress the Equilibrium Rule. Crucially, the controlling parameter values in every case.

TABLE I. Composition of the Structural Review Test Cases v3 benchmark constructed from real-project BIM exports

Category	Number of cases	Description
Compliant	66	All parameters satisfy every applicable clause.
Single violation	65	Exactly one parameter violates one applicable benchmark clause.
Compound violation	33	Two or more parameters violate simultaneously.
Boundary	27	Parameters sit at the numerical boundary of a clause.
Incomplete information	20	One or more required parameters are null (reflecting genuine BIM-export omissions).
Total	211	

Ground-truth labels (APPROVE, REJECT, and NEED-MORE-INFO) and the set of triggered rules are finalized through expert review against the applicable benchmark clauses extracted from GB 50010 and GB 50009, with a deterministic script used only for consistency checking. The parameter values come directly from real BIM exports, and Figure 3 reports the overall accuracy of all evaluated systems.

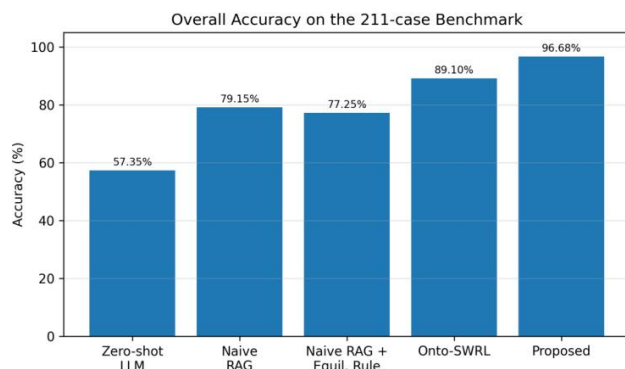


FIGURE 3. Overall accuracy of the evaluated systems on the 211-case benchmark.

We compare five systems on the same 211-case bench-

mark, all using GPT-5.4 as the generation LLM (accessed through an OpenAI-compatible forwarding endpoint) and Qwen3-Embedding-0.6B as the embedding model.

Zero-Shot LLM. The raw query, with no retrieval, is passed to the LLM together with a short system prompt describing the output format.

Naive RAG. Top-5 dense retrieval over the raw corpus, no corpus augmentation, single-agent generation.

Naive RAG + Equilibrium Rule (single-agent). Identical retrieval to the Naive RAG baseline, but the generation prompt is augmented with the Equilibrium Rule (Section 3.3).

Onto-SWRL (re-implemented baseline). We re-implement a neural-symbolic compliance-checking pipeline in which the input is first routed through an LLM that produces an OWL-style JSON assertion, and the assertion is then evaluated by a deterministic reasoner that implements a subset of GB 50010 as SWRL-like rules. The implementation covers clauses that are naturally expressible in the Web Ontology Language Description Logic (OWL-DL) subset commonly used in this literature; arithmetic-ratio and multi-condition rules remain outside that schema. The resulting baseline reaches 89.10% overall accuracy on the 211-case benchmark and serves as a symbolic reference point for this task setting.

Proposed (full role-specialized system). The tripartite pipeline described in Section 3 with the augmented retrieval corpus of Section 3.4 (24 snippets).

B. OVERALL PERFORMANCE AND DETAILED ACCURACY

Under the reported configuration, the proposed framework attains the highest accuracy on this benchmark and is the only method in this comparison above 95% accuracy. Relative to Naive RAG, the gain is concentrated in cases where the controlling evidence is distributed across clauses or where the final decision benefits from a structured verdict layer. As shown in Figure 4 and Table II–III, the Naive RAG + Equilibrium Rule variant indicates that the prompt rule alone is insufficient in a single-stage pipeline, while the Onto-SWRL baseline remains competitive but weaker on arithmetic-ratio and cross-condition cases. Because the proposed system bundles role-specialized reasoning, retrieval augmentation, and a larger retrieval budget, these results should be interpreted as system-level comparisons; the ablations below provide partial evidence about component-level contributions within that system.

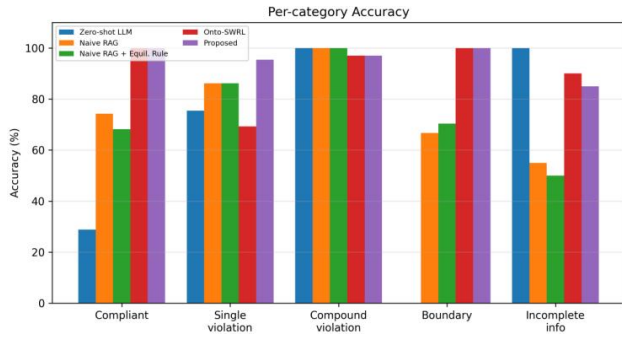


FIGURE 4. Per-category accuracy of the evaluated systems.

Several trends are worth highlighting. First, the zero-shot LLM scores 100% on incomplete-information cases largely because it defaults to NEED-MORE-INFO for many non-trivial inputs. Second, the Naive RAG baseline loses substantial accuracy on compliant and incomplete-information cases, indicating that retrieval alone is insufficient for stable structural-review decisions. Third, adding the Equilibrium Rule to Naive RAG (single-agent) does not recover compliant-case accuracy and slightly hurts incomplete-information accuracy. Fourth, the Onto-SWRL baseline is weakest on single-violation cases, where arithmetic and cross-condition rules appear more frequently. The proposed framework is also the only system in this comparison that achieves at least 93% on every component class. Naive RAG is particularly weak on slabs (62.5%), which is consistent with the heavier reliance of slab rules on environmental-class and cover interactions. Adding the Equilibrium Rule to a single-agent Naive RAG raises slab accuracy modestly (+3.75 pt) but reduces column accuracy (-7.69 pt), indicating that prompt guidance alone does not stabilize all component types equally.

The remaining 7 errors of the proposed framework are concentrated in the incomplete-information category (3 errors) and in single- or compound-violation cases where the controlling clause involves a multi-step arithmetic derivation (4 errors). Type-I errors (false approvals) account for 3 of the 7 cases, and Type-II errors (over-rejections) account for the remaining 4. Tables II–IV and Figure 5 report the accuracy details by method, category, and component type [30].

TABLE II. Overall accuracy with Wilson 95% confidence intervals ($n = 211$ for every row)

Method	Correct	Total	Accuracy	Wilson 95% CI
Zero-Shot LLM	121	211	57.35%	[50.61, 63.83]
Naive RAG	167	211	79.15%	[73.17, 84.08]
Naive RAG + Equilibrium Rule (single-agent)	163	211	77.25%	[71.14, 82.39]
Onto-SWRL (re-implemented)	188	211	89.10%	[84.17, 92.63]
Proposed Framework	204	211	96.68%	[93.31, 98.38]

TABLE III. Per-category accuracy (percent) on the 211-case benchmark

Category	Zero-Shot	Naive RAG	Naive RAG + Equilibrium Rule	Onto-SWRL	Proposed
Compliant ($n = 66$)	28.79%	74.24%	68.18%	100.00%	100.00%
Single violation ($n = 65$)	75.38%	86.15%	86.15%	69.23%	95.38%
Compound violation ($n = 33$)	100.00%	100.00%	100.00%	96.97%	96.97%
Boundary ($n = 27$)	0.00%	66.67%	70.37%	100.00%	100.00%
Incomplete info ($n = 20$)	100.00%	55.00%	50.00%	90.00%	85.00%

TABLE IV. Accuracy by component type (percent)

Component	Zero-Shot	Naive RAG	Naive RAG + Equilibrium Rule	Onto-SWRL	Proposed
Slab ($n = 80$)	53.75%	62.50%	66.25%	86.25%	98.75%
Beam ($n = 66$)	62.12%	92.42%	89.39%	89.39%	93.94%
Column ($n = 65$)	56.92%	86.15%	78.46%	92.31%	96.92%

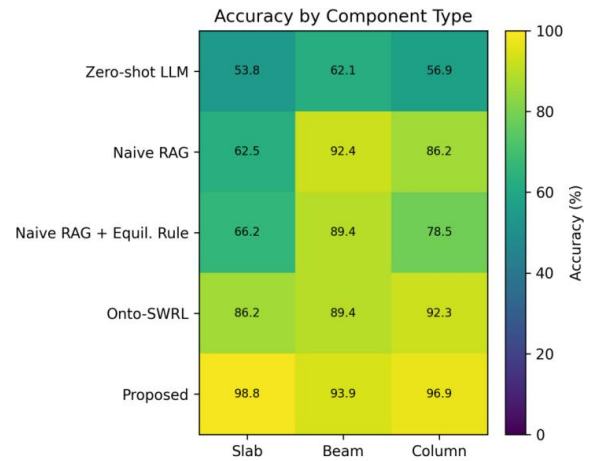


FIGURE 5. Accuracy heatmap across the three component types.

C. ABLATION, DIAGNOSTICS AND SIGNIFICANCE

The ablation study provides a clear empirical message. Removing the Equilibrium Rule or the Critic leads to a large drop, indicating that stable decision behavior and stable verdict extraction both matter in this task setting. Removing corpus augmentation produces a smaller decrease, with most of the loss concentrated in compliant and boundary cases (Table V–VII).

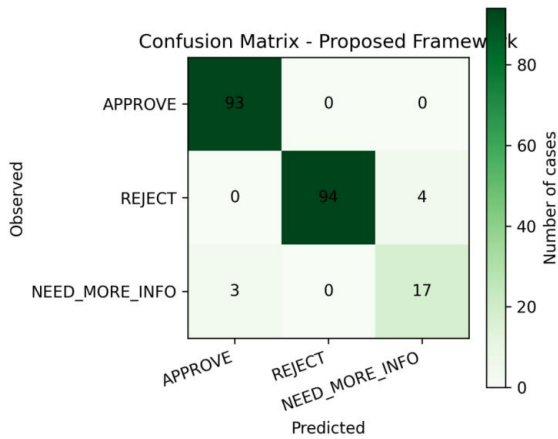


FIGURE 6. Confusion matrix of the proposed framework on the 211-case benchmark.

Figure 6 displays the confusion matrix of the proposed framework and shows that the residual errors are concentrated in incomplete-information and violation cases. We therefore interpret augmentation as a helpful retrieval aid, but not the primary driver of the final result on this benchmark.

The one-factor and two-factor ablation results are reported below. These ablations clarify component importance within the proposed system, but they do not fully orthogonalize every factor in the headline comparison against Naive RAG, because the proposed system also differs in retrieval corpus and top-k setting. We therefore treat the ablation evidence as informative but not as a perfectly clean decomposition of architecture, retrieval patching, and retrieval-budget effects.

TABLE V. One-factor ablation of the three proposed components ($n = 211$)

Variant	Correct	Total	Accuracy	Δ vs. Full
Full (Proposed)	204	211	96.68%	—
w/o corpus augmentation	200	211	94.79%	-1.89
w/o Critic	172	211	81.52%	-15.16
w/o Equilibrium Rule	158	211	74.88%	-21.80

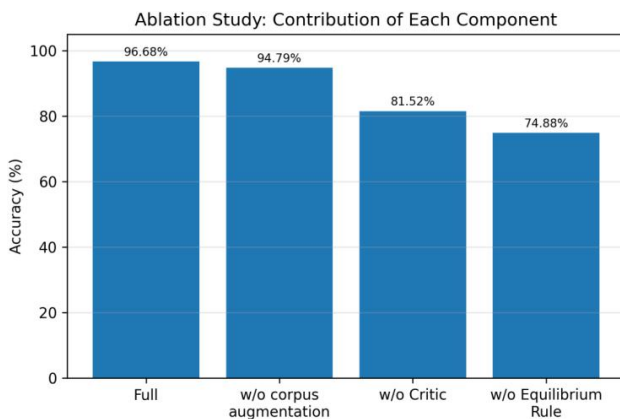


FIGURE 7. Contribution of each component of the proposed framework. All numbers are evaluated on the same 211-case benchmark.

Figure 7 visualizes the contribution of each proposed component and confirms that the Equilibrium Rule and the Critic account for the largest drops when removed.

TABLE VI. 2×2 factorial ablation, as-deployed defaults ($n = 211$). Corpus augmentation is ON in the role-specialized row and OFF in the Naive RAG row

System	w/o Equilibrium Rule	w/ Equilibrium Rule
Naive RAG (single-agent, raw corpus only)	79.15% (167/211)	77.25% (163/211)
Role-specialized pipeline (Parser+Reviewer+Critic, augmented corpus)	74.88% (158/211)	96.68% (204/211)

TABLE VII. Purified 2×2 factorial ablation with corpus augmentation OFF in every cell

System	w/o Equilibrium Rule	w/ Equilibrium Rule
Naive RAG	79.15% (167/211)	77.25% (163/211)
Role-specialized pipeline	69.19% (146/211)	94.79% (200/211)

Figure 8 provides the additional diagnostic heat-map for the 2×2 factorial ablation. This check is included as supporting evidence rather than as a central claim of the paper, and it shows that neither the staged architecture alone nor the prompt rule alone gives the best result, whereas the full combination does.

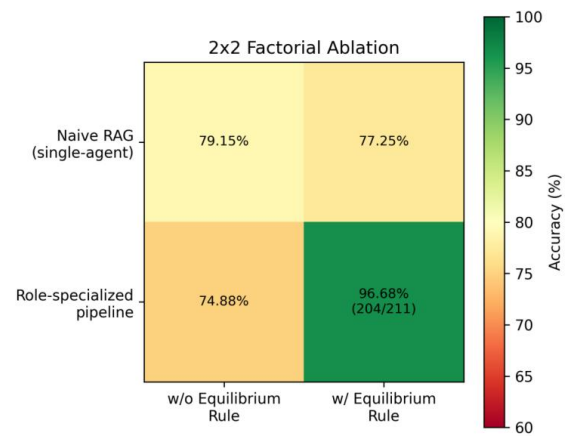


FIGURE 8. Heat-map visualization of the as-deployed 2×2 factorial ablation (Table VI). The full combination performs best in both the as-deployed and purified variants.

The McNemar exact tests [31] support the same overall conclusion as the accuracy results: the proposed framework outperforms the major baselines on this benchmark, whereas the marginal effect of corpus augmentation is not statistically established at the current sample size. We include these tests as supporting evidence while keeping the main emphasis on empirical performance in a realistic task setting.

D. ERROR ANALYSIS

The proposed framework makes 7 errors in total. Three occur in incomplete-information cases, where missing fields still create ambiguity for the downstream reasoning stages. The other four occur in single- or compound-violation cases involving multi-step arithmetic derivations, which remain difficult even when the relevant clauses are retrieved. Overall, the residual failure modes are concentrated in numeric reasoning and incomplete input handling rather than in

simple threshold matching.

From a safety-oriented perspective, the 7 residual errors split into 3 false approvals and 4 over-rejections. This corresponds to a false-approval rate of 1.42% over all 211 cases and highlights that the headline overall accuracy should be read together with the error profile, especially in a structural-review setting where false approvals are more consequential than over-rejections (Table VIII).

TABLE VIII. McNemar exact tests, Proposed vs. each baseline ($n = 211$)

Baseline	b (Prop✓, Base×)	c (Prop×, Base✓)	Δ accuracy (pt)	Exact two-sided p
Zero-Shot LLM	89	6	+39.34	$< 10^{-4}$
Naive RAG	41	4	+17.54	$< 10^{-4}$
Naive RAG + Equilibrium Rule	45	4	+19.43	$< 10^{-4}$
Onto-SWRL (re-implemented)	20	4	+7.58	0.0015
w/o Equilibrium Rule	53	7	+21.80	$< 10^{-4}$
w/o Critic	39	7	+15.17	$< 10^{-4}$
w/o corpus augmentation	10	6	+1.90	0.4545

V. CONCLUSION

This paper presents a role-specialized RAG framework for automated structural compliance checking and evaluates it on a 211-case benchmark derived from real-project BIM exports and expert-reviewed labels. The study draws its benchmark knowledge resources, retrieval corpus, and rule-supporting mappings from GB 50010 and GB 50009, although the present task remains centered on reinforced-concrete member review. The framework combines a deterministic Parser, a Reviewer guided by the Equilibrium Rule, a Critic for structured verdict extraction, and lightweight retrieval-corpus augmentation. Under the reported implementation, it achieves 96.68% overall accuracy and outperforms the zero-shot LLM, Naive RAG, Naive RAG with the Equilibrium Rule, and re-implemented Onto-SWRL baselines evaluated in this study.

These results suggest that an effective engineering compliance-review system can be built from a modest set of well-defined components rather than from a single monolithic prompting strategy or a fully formalized ontology workflow. The main value of the framework lies in its auditable division of labor: the Parser stabilizes retrieval-query construction, the Reviewer grounds reasoning in retrieved clauses, and the Critic converts reasoning into a structured decision-support output. Nevertheless, the findings should be read with explicit boundaries. The benchmark covers one national regulatory setting, one language, one generation model, and one embedding model; the retrieval settings are not perfectly identical across all baselines; and the corpus-augmentation effect is directionally positive but not statistically established at the current sample size.

For safety-critical deployment, the framework should not

be treated as a fully autonomous approval engine. The Equilibrium Rule is an engineering heuristic for this task setting, not a general legal or safety doctrine, and the residual false approvals show that engineers must retain final responsibility. Future research should extend the evaluation to additional codes, jurisdictions, component types, and LLM families; strengthen arithmetic and multi-condition reasoning; test independent labeling and retrieval settings; and report cost-sensitive metrics such as false-approval rate and class-wise error profiles.

REFERENCES

- [1] J. Zhang et al., "Factors influencing the acceptance of BIM-based automated code compliance checking in the AEC industry in China," *J. Manage. Eng.*, vol. 39, no. 5, Art. no. 04023036, 2023.
- [2] J. Peng and X. Liu, "Automated code compliance checking research based on BIM and knowledge graph," *Sci. Rep.*, vol. 13, Art. no. 7065, 2023.
- [3] R. Zhang, "BIM, NLP, and AI for automated compliance checking," in *Proc. Comput. Civil Eng.*, Reston, VA, USA, 2022.
- [4] C. Zhang and N. M. El-Gohary, "Integrating semantic NLP and logic reasoning into a unified system for fully automated code checking," *Autom. Constr.*, vol. 73, pp. 45–57, 2017.
- [5] S. Li et al., "Automated compliance checking for BIM models based on Chinese NLP and knowledge graph: An integrative conceptual framework," *Eng. Constr. Archit. Manag.*, 2024.
- [6] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Proc. Adv. Neural Inf. Process. Syst.* 33, Red Hook, NY, USA, 2020.
- [7] J. K. Lee et al., "High-level implementable methods for automated building code compliance checking," *Dev. Built Environ.*, vol. 15, Art. no. 100174, 2023.
- [8] N. Chen et al., "Automated building information modeling compliance check through a large language model combined with deep learning and ontology," *Buildings*, vol. 14, no. 7, Art. no. 1983, 2024.
- [9] J. Peng and X. Liu, "Knowledge-graph-driven automated drawing review and BIM compliance checking," *Sci. Rep.*, vol. 13, Art. no. 7065, 2023.
- [10] S. Fuchs et al., "Using large language models for the interpretation of building regulations," *arXiv preprint arXiv:2407.21060*, 2024. [Online]. Available: <https://arxiv.org/abs/2407.21060>.

- [11] S. Madireddy et al., “Large language model-driven code compliance checking in building information modeling,” *Electronics*, vol. 14, no. 11, Art. no. 2146, 2025.
- [12] X. Xu and H. Cai, “Ontology and rule-based natural language processing approach for interpreting textual regulations on underground utility infrastructure,” *Adv. Eng. Inform.*, vol. 48, Art. no. 101288, 2021.
- [13] M. Aydin, “Building information modeling based automated building regulation compliance checking ASP.NET web software,” *Intell. Autom. Soft Comput.*, vol. 28, no. 1, pp. 11–25, 2021.
- [14] Z. Zhang et al., “Towards fully automated code compliance checking of building regulations: challenges for rule interpretation and representation,” in *Proc. 2022 Eur. Conf. Comput. Constr.*, Rhodes, Greece, 2022.
- [15] D. Kampelopoulos et al., “A review of LLMs and their applications in the architecture, engineering and construction industry,” *Artif. Intell. Rev.*, vol. 58, no. 8, Art. no. 250, 2025.
- [16] Y. Zheng, “Evaluating the potential of large language models for building information modeling macro generation,” *arXiv preprint arXiv:2307.11623*, 2023. [Online]. Available: <https://arxiv.org/abs/2307.11623>.
- [17] S. Madireddy et al., “Large language model-driven code compliance checking in building information modeling,” *Electronics*, vol. 14, no. 11, Art. no. 2146, 2025.
- [18] G. Izacard and E. Grave, “Leveraging passage retrieval with generative models for open-domain question answering,” in *Proc. 16th Conf. Eur. Chapter Assoc. Comput. Linguist.*, Stroudsburg, PA, USA, 2021, pp. 874–880.
- [19] L. Gao et al., “Precise zero-shot dense retrieval without relevance labels,” in *Proc. 61st Annu. Meet. Assoc. Comput. Linguist.*, Stroudsburg, PA, USA, 2023, pp. 1762–1777.
- [20] L. Wang et al., “Query2Doc: Query expansion with large language models,” in *Proc. Conf. Empir. Methods Nat. Lang. Process.*, Stroudsburg, PA, USA, 2023.
- [21] A. Asai et al., “Self-RAG: Learning to retrieve, generate, and critique through self-reflection,” in *Proc. Int. Conf. Learn. Represent.*, 2024.
- [22] Y. Zhou et al., “Trustworthiness in retrieval-augmented generation systems: A survey,” *arXiv preprint arXiv:2409.10102*, 2024. [Online]. Available: <https://arxiv.org/abs/2409.10102>.
- [23] J. Lee et al., “Performance comparison of retrieval-augmented generation and fine-tuned large language models for construction safety management knowledge retrieval,” *Autom. Constr.*, vol. 168, Art. no. 105846, 2024.
- [24] J. Wei et al., “Chain-of-thought prompting elicits reasoning in large language models,” in *Proc. Adv. Neural Inf. Process. Syst. 35*, Red Hook, NY, USA, 2022.
- [25] D. Zhou et al., “Least-to-most prompting enables complex reasoning in large language models,” in *Proc. Int. Conf. Learn. Represent.*, 2023.
- [26] T. Kojima et al., “Large language models are zero-shot reasoners,” in *Proc. Adv. Neural Inf. Process. Syst. 35*, Red Hook, NY, USA, 2022.
- [27] Q. Wu et al., “AutoGen: Enabling next-generation LLM applications via multi-agent conversation,” *arXiv preprint arXiv:2308.08155*, 2023. [Online]. Available: <https://arxiv.org/abs/2308.08155>.
- [28] S. Hong et al., “MetaGPT: Meta programming for a multi-agent collaborative framework,” in *Proc. Int. Conf. Learn. Represent.*, 2024.
- [29] X. Liu et al., “AgentBench: Evaluating LLMs as agents,” in *Proc. Int. Conf. Learn. Represent.*, 2024.
- [30] L. D. Brown, T. T. Cai and A. Dasgupta, “Interval estimation for a binomial proportion,” *Stat. Sci.*, vol. 16, no. 2, pp. 101–133, 2001.
- [31] M. W. Fagerland et al., “Statistical analysis of contingency tables,” Boca Raton, FL, USA: CRC Press, 2017.