EVIDENCE-GROUNDED RETRIEVAL OF HISTORICAL BIM COORDINATION KNOWLEDGE: A HUMAN-IN-THE-LOOP FRAMEWORK AND OPEN SYNTHETIC BENCHMARK

Roja Malligarjunan, M.S.
Independent researcher, Lynnwood, Washington, USA
https://orcid.org/0009-0002-2480-6722
roja6055@gmail.com

Preprint, version 1. This manuscript has not been peer reviewed.

SUMMARY: *On repeat programs such as data-center campuses, the same coordination clashes recur across projects that share a basis of design, yet each project team resolves them again because resolution knowledge stays inside individual project records. Large language models make it easy to summarize old records, but an ungrounded answer to an engineering question is a liability. This article proposes a human-in-the-loop framework for retrieving historical coordination precedents and an open benchmark for testing it. The framework contributes four elements. First, an issue-record schema keeps the engineering context of a clash, such as grid zone, disciplines and elements, alongside its text. Second, a hybrid retrieval method fuses keyword, semantic and structured-feature rankings, so that two issues with similar wording but different root causes are separated, and demotes precedents that a later design revision has superseded. Third, an answer contract allows a generated summary to state only what cited records support and requires it to say when they support nothing. Fourth, a feedback loop turns each reviewer's acceptance or rejection into labelled data for calibration. On a synthetic benchmark of 200 issue records and 40 queries with deliberate hard negatives, keyword and semantic retrieval placed 6.7% to 9.2% hard negatives and 10.8% to 13.3% stale precedents in their top five results on the held-out split. The hybrid with superseded-record demotion raised nDCG@5 from 0.825 for keyword retrieval to 0.915, cut hard negatives to 2.5% and stale precedents to 5.8%, and abstained correctly on all three held-out queries without a precedent. Demotion gave the largest single gain; the remaining stale results were supersessions that no one had recorded. The answer contract kept 92% of cited statements supported, against 44% for an unconstrained summary. These are results on synthetic data, not evidence of field accuracy. The benchmark and code are released openly.*

KEYWORDS: *BIM coordination, clash resolution, knowledge reuse, hybrid retrieval, retrieval-augmented generation*

1. INTRODUCTION

Owners who build to a standard basis of design construct the same building many times. Data-center operators, logistics companies, retailers and health-care systems all deliver portfolios of facilities from a common set of design standards, often with the same consultants and trade contractors. Repetition should make design coordination faster. In practice, a clash between a cable tray and a duct at a given grid intersection is investigated again on each project, even when the same clash, with the same cause, was resolved months earlier on a sister building.

The reasons are organizational rather than technical. Issues live in per-project coordination databases. The useful part of a resolution sits in review comments, markups and engineering advice rather than in a structured field. The people who resolved the issue move to other projects. Construction research has documented for more than a decade that lessons-learned processes capture knowledge but rarely deliver it to the next project at the moment it is needed (Carrillo et al., 2013).

Retrieval-augmented generation (RAG), which conditions a language model on documents retrieved for a query (Lewis et al., 2020), appears to solve this problem: index past issues, retrieve the closest ones for a new clash, and let a language model summarize what was done. Three properties of coordination knowledge make a naive implementation risky.

1. **Words mislead.** Two issues described as "tray clashes with duct" can have unrelated causes, such as a redesigned busway route in one case and a raised ceiling in the other. Location and element context distinguish them; text alone does not.

2. **Precedents go stale.** A resolution adopted before a basis-of-design revision can be wrong after it, even though it remains the closest textual match.
3. **Answers carry authority.** A fluent summary of a weak match reads like advice. In engineering coordination, a system should locate evidence for a qualified person and should never decide.

This article proposes a framework built around those three properties and tests it on an open synthetic benchmark. Its contributions are:

1. an issue-record schema that separates engineering context, narrative and governance fields;
2. a hybrid retrieval method that fuses keyword, semantic and structured-feature rankers by weighted reciprocal rank fusion and demotes superseded precedents;
3. an answer contract for any generated summary, including an explicit “no support” outcome;
4. a feedback loop in which reviewer decisions become relevance labels; and
5. an open benchmark of 200 synthetic issue records and 40 queries with built-in hard negatives, released with code and run files.

The work is motivated by the author’s practice as a BIM information manager on large repeat programs, where investigating a recurring clash from scratch can take hours. No data from that practice are used here; the evaluation relies entirely on an open synthetic benchmark.

Section 2 reviews related work. Section 3 presents the framework. Section 4 describes the benchmark and evaluation protocol, and Section 5 reports results. Section 6 discusses implications for practice, limitations and threats to validity, and Section 7 concludes.

2. RELATED WORK

2.1 Lessons learned and case retrieval in construction

Knowledge management research in construction has long distinguished capturing lessons from reusing them. Carrillo et al. (2013) found that lessons-learned practice in construction organizations captures knowledge at project close-out but seldom delivers it to later projects at the point of need. Case-based approaches address delivery by retrieving similar past cases for a new situation. Zou et al. (2017) combined a vector space model with semantic query expansion to retrieve similar cases for construction risk management from plain-language reports. Their work shows that text retrieval can surface relevant precedents from large case bases, and also that text similarity alone is the primary signal in such systems.

2.2 Computational support for clash management

Research on BIM clash management has focused mainly on which clashes matter and how to resolve them. Hu and Castro-Lacouture (2019) used machine learning on clash attributes to predict clash relevance, showing that engineering features of a clash carry information that its description does not. Hsu et al. (2020) built a knowledge-based system that combines machine learning of constructors’ resolution knowledge with simulated annealing to propose clash resolutions before coordination meetings. These studies treat clash features as inputs to classification or optimization. This article uses similar engineering features for a different purpose: to find resolved precedents and present them, with their sources, to the engineer who must decide.

2.3 Retrieval-augmented language models in construction

Since 2023, construction researchers have applied RAG to several document-heavy tasks. Tran et al. (2024) built a retrieval-augmented assistant over construction safety regulations and reported favourable user ratings while noting open concerns about retrieval precision and hallucinated answers. Uhm et al. (2025) compared a retrieval-augmented system with general-purpose language models for construction safety information and found that retrieval improved accuracy and relevance as judged by experts and workers. Wu et al. (2025) proposed RAG4CM, which organizes project documents into hierarchical knowledge pools for question answering in construction management. Xia et al. (2026) emphasized traceability of language-model outputs to construction elements in report generation. Together these studies show that grounding improves answer quality and that traceability matters to practitioners. They retrieve mainly by text similarity over regulations or project documents. Coordination issue records differ: they are short, repetitive in vocabulary, and distinguished chiefly

by location, discipline and element context, so a text-only retriever is most likely to return the wrong precedent with confidence.

2.4 Ranking and evaluation methods

The method builds on standard information-retrieval components. BM25 remains a strong keyword baseline (Robertson & Zaragoza, 2009). Latent semantic analysis provides a light, reproducible semantic baseline that captures paraphrase without external models (Deerwester et al., 1990). Reciprocal rank fusion combines rankings from different systems robustly without calibrating their score scales (Cormack et al., 2009). Normalized discounted cumulative gain (nDCG) evaluates ranked lists against graded relevance judgments (Järvelin & Kekäläinen, 2002), and heterogeneous benchmarks such as BEIR have shown the value of shared test collections for comparing retrievers (Thakur et al., 2021).

2.5 Gap

No published method or shared benchmark, to the author's knowledge, addresses retrieval of coordination precedents in a way that (a) treats engineering context as a first-class ranking signal, (b) handles precedents made stale by design revisions, (c) separates retrieval quality from the faithfulness of any generated summary, and (d) keeps the engineer's decision explicit and recorded. Table 1 summarizes how this article relates to the closest prior work.

Table 1: Closest prior work and the contribution of this article.

Prior work	What it establishes	What this article adds
Lessons-learned practice (Carrillo et al., 2013)	Knowledge is captured but rarely reused at the point of need	Retrieval triggered by a new issue inside the coordination workflow
Case retrieval with NLP (Zou et al., 2017)	Text retrieval can surface similar past cases	Engineering context fused with text to separate look-alike cases
Clash relevance and resolution (Hu & Castro-Lacouture, 2019; Hsu et al., 2020)	Clash features predict relevance; resolution knowledge can be learned	Uses clash features to retrieve resolved precedents for a human decision
RAG in construction (Tran et al., 2024; Uhm et al., 2025; Wu et al., 2025; Xia et al., 2026)	Grounding and traceability improve answers	Answer contract restricted to cited coordination records with an explicit "no support" outcome
Rank fusion and benchmarks (Cormack et al., 2009; Thakur et al., 2021)	Robust fusion; shared test collections	A small open domain benchmark with hard negatives specific to coordination

3. THE FRAMEWORK

Figure 1 shows the framework. A new clash issue, with its engineering context, is scored against every resolved record by three rankers. Their rankings are fused, superseded records are demoted, and the ranked precedents are shown to an engineer with their record identifiers. An optional summary obeys the answer contract. The engineer's decision is stored with the record, so every use of the system adds labelled data.

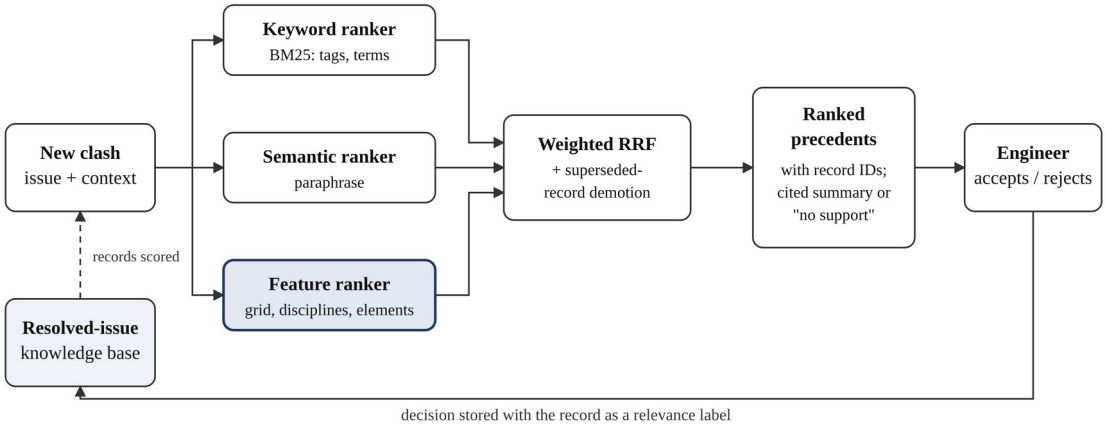


Figure 1: Framework for evidence-grounded retrieval of coordination precedents.

3.1 Issue-record schema

Each resolved issue is stored as a record with three kinds of field (Table 2). Context fields let the system tell apart issues that read alike. Narrative fields carry what a reviewer needs to judge applicability. Governance fields let the system cite, filter by permission, and handle stale precedents.

Table 2: Issue-record schema.

Field group	Fields	Purpose
Context	Project type, phase, basis-of-design family and version, grid zone, level, discipline pair, element categories, clash type	Separates issues with similar wording but different causes
Narrative	Symptom, root cause, resolution, reviewer comments	Lets a reviewer judge whether the precedent applies
Governance	Record ID, source document, resolution date, superseded-by link, quality label, permission label	Supports citation, access control and demotion of stale records

3.2 Hybrid retrieval

Three rankers score every record against the new issue. A keyword ranker (BM25) matches exact equipment tags and terms (Robertson & Zaragoza, 2009). A semantic ranker matches paraphrase; in the released version it is latent semantic analysis over the narrative fields (Deerwester et al., 1990), chosen because it is light and fully reproducible, and it can be replaced by a dense embedding model. A structured-feature ranker scores agreement on grid zone, discipline pair (in either order), element categories, clash type, phase and basis-of-design family.

The three ranked lists are combined by weighted reciprocal rank fusion (Cormack et al., 2009), which needs no calibration between the rankers' score scales:

$$\mathrm{RRF}(d) = \sum_{r \in \{\mathrm{kw},\ \mathrm{sem},\ \mathrm{feat}\}} \frac{w_r}{k + \mathrm{rank}_r(d)}, \qquad k = 60 \tag{1}$$

where $\mathrm{rank}_r(d)$ is the position of record d in ranker r's list and w_r is the weight of ranker r. With all weights equal to 1, Equation 1 is standard reciprocal rank fusion.

Records marked superseded, or resolved under a basis-of-design version older than the project's, are moved below all current records and flagged rather than hidden, because a stale precedent can still explain a cause.

Demotion can only act on supersessions that were recorded; Section 5 measures what remains when they were not.

3.3 Answer contract

An optional language-model layer may summarize the top records. It operates under four rules:

1. every sentence cites at least one record ID;
2. it may not introduce a cause or fix that is absent from the cited records;
3. when no record supports an answer, it says so and returns the nearest records as leads only; and
4. it never phrases output as an instruction.

Retrieved records are always shown beside the summary, so a reviewer can check every statement. A record may be cited only when its structured-feature agreement with the new issue reaches a threshold tuned on development queries, and the system abstains when no record does.

3.4 Presentation tiers, feedback and calibration

Fusion scores are ranks, not probabilities. The interface therefore shows tiers (suggested, suggested with caution, informational) rather than confidence values. The tier cut-offs should be set from reviewer acceptance rates observed in a shadow-mode pilot, in which suggestions are logged but not shown. Until that calibration exists, no score is presented as confidence. Each acceptance or rejection is stored with the record and the query, which turns routine use into relevance judgments for tuning the weights in Equation 1.

4. BENCHMARK AND EVALUATION PROTOCOL

4.1 Research questions

1. RQ1. Does hybrid retrieval with the structured-feature ranker rank relevant precedents higher than keyword-only, semantic-only, and keyword-plus-semantic baselines, and does it keep hard negatives out of the top results?
2. RQ2. Does the answer contract produce only supported, cited statements and abstain when no precedent exists?
3. RQ3. How much does performance depend on shared grid naming and basis of design between the new project and the historical records?

4.2 Benchmark construction (version 1.0)

All data are synthetic. No employer issue records, prompts or knowledge bases were used. The benchmark is generated by a published script with a fixed seed, so it can be rebuilt exactly. It contains 200 issue records following the schema in Table 2, across architecture, structure, mechanical, electrical, plumbing and fire protection: 12 recurring scenarios with 14 records each, and 32 unrelated records.

Three pairs of scenarios share symptom wording, basis-of-design family and neighbouring grid zones, but differ in disciplines, elements and root cause; their records are the hard negatives for each other’s queries. Each scenario also has two stale records whose resolution no longer applies: one is recorded as superseded, and the other’s supersession was never recorded, as often happens in practice.

To mimic the noise of real issue logs, 35% of records have no recorded cause, 15% have a reversed or missing discipline pair, 35% sit at a neighbouring or random grid zone, 20% have incomplete element categories, 30% list a different phase and 15% a different level. Queries are noisy too: 40% give a neighbouring grid zone, 30% give only one element category, 30% give a different phase, and half the time a term is replaced by a synonym that records do not use (for example “cable ladder” for “cable tray”).

There are 40 queries: 13 for tuning (development) and 27 for reporting (test). Four queries have no precedent in the records; three of these are in the test split. Relevance labels are derived from the generation process: current records of the query’s scenario are graded 3, its stale records 1, and all others 0; records of the look-alike scenario are marked as hard negatives. Independent human relevance judgments are planned for version 1.1.

4.3 Methods compared and parameters

Seven configurations were compared: keyword (BM25, $k_1 = 1.2$, $b = 0.75$); semantic (latent semantic analysis, 40 dimensions); keyword plus semantic fused by reciprocal rank fusion; feature only; hybrid with equal weights; hybrid with superseded-record demotion; and weighted hybrid with demotion. The structured-feature score sums agreement on grid zone (1 for the same zone, 0.5 for a neighbour), discipline pair (1, or 0.5 for one shared discipline), element categories (Jaccard overlap), basis-of-design family (1), clash type (0.5), phase (0.5) and level (0.25), for a maximum of 4.75.

Parameters were tuned on the development split only. RRF k was fixed at 60. The feature-ranker weight was chosen from 1 to 5 by development nDCG@5 (0.884, 0.920, 0.941, 0.939 and 0.936), giving $w_{feat} = 3$ with keyword and semantic weights of 1. The abstention and citation threshold was set midway between the best feature score of the development query without a precedent and the lowest best score of the development queries with one, giving 4.0.

To test the answer contract without the variability of a particular language model, the summary layer was simulated by rules that apply the contract exactly: one statement per cited record, reporting only the cause and fix written in that record. The baseline is an unconstrained summary of the top three keyword results that always answers and fills a missing cause with the cause of the highest-ranked record that has one, a typical unsupported inference. Paired differences in nDCG@5 were tested with a bootstrap interval and the Wilcoxon signed-rank test.

4.4 Metrics

Table 3 lists the metrics. nDCG@5 and mean reciprocal rank (MRR) measure ranking quality against graded labels (Järvelin & Kekäläinen, 2002); 95% intervals for nDCG@5 were obtained by bootstrap resampling of test queries.

Table 3: Evaluation metrics.

Metric	Measures	Question
nDCG@5, MRR	Ranking quality against graded labels	RQ1, RQ3
Recall@10	Share of relevant records found in the top 10	RQ1
Hard negatives in top 5	Share of top-5 results from a look-alike scenario	RQ1
Stale in top 5	Share of top-5 results whose resolution was superseded	RQ1
Citation precision	Share of cited records that support the statement	RQ2
Correct abstention	“No support” returned when no precedent exists, and only then	RQ2

5. RESULTS

Table 4 and Figure 2 report results on the held-out test split for the 24 queries that have a precedent.

Table 4: Results on the held-out test split (24 queries with a precedent); 95% bootstrap interval for nDCG@5 in brackets.

Method	nDCG@5	MRR	Recall@10	Hard negatives in top 5	Stale in top 5
Keyword (BM25)	0.825 (0.744–0.890)	0.904	0.622	9.2%	10.8%
Semantic (LSA)	0.806 (0.730–0.873)	0.885	0.628	8.3%	12.5%

Method	nDCG@5	MRR	Recall@10	Hard negatives in top 5	Stale in top 5
Keyword + semantic	0.821 (0.744–0.880)	0.885	0.618	6.7%	13.3%
Feature only	0.880 (0.812–0.937)	0.938	0.708	3.3%	10.8%
Hybrid, equal weights	0.863 (0.809–0.913)	0.938	0.681	4.2%	12.5%
Hybrid + superseded demotion	0.916 (0.860–0.965)	0.958	0.729	5.0%	5.0%
Weighted hybrid + demotion	0.915 (0.845–0.970)	0.917	0.764	2.5%	5.8%

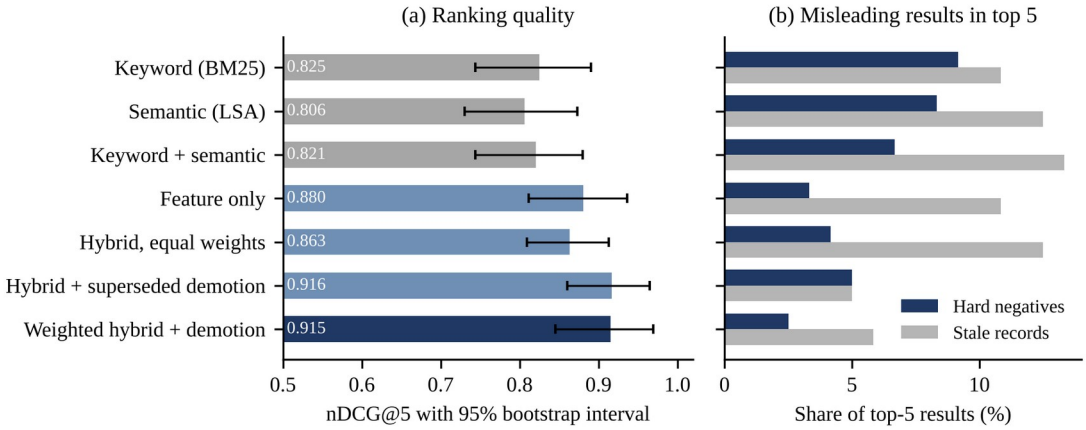


Figure 2: (a) nDCG@5 with 95% bootstrap intervals; (b) share of hard negatives and stale records among the top five results.

RQ1. Text-only rankers were misled by shared wording and stale precedents: keyword, semantic and combined retrieval placed 6.7% to 9.2% hard negatives and 10.8% to 13.3% stale records in their top five results. The structured-feature ranker cut hard negatives to 3.3% on its own, but it could not tell stale precedents from current ones. Demotion made the largest single difference: adding it to the equal-weight hybrid raised nDCG@5 by 0.053 (95% interval 0.025 to 0.082; Wilcoxon p = 0.007) and halved the stale share to 5.0%. The weighted hybrid with demotion improved nDCG@5 over keyword retrieval by 0.090 (0.011 to 0.180; p = 0.049) and over keyword plus semantic by 0.094 (0.012 to 0.179; p = 0.031), had the fewest hard negatives (2.5%) and the highest recall@10 (0.764). Weighting the feature ranker did not change nDCG@5 relative to the equal-weight hybrid with demotion (difference −0.001, p = 0.67). The stale records that remained in the top five were the supersessions that had never been recorded, which no ranking step can detect.

RQ2. Under the answer contract, 65 of 71 cited statements (92%) were supported by a current precedent of the right scenario; the six unsupported citations were records that passed the threshold but belonged to a look-alike scenario or were unrecorded supersessions. The unconstrained summary produced 81 statements, of which 36 (44%) were supported, and it answered all three queries that had no precedent. The contract abstained on all three of those queries and on none that had a precedent.

RQ3. When the new project shared neither grid naming nor basis of design with the historical records, nDCG@5 for the weighted hybrid fell from 0.915 to 0.896, still above every text-only method. Removing one feature group at a time showed that discipline pair (0.904 without it) and element categories (0.909) carried the most information. Removing grid zone or basis-of-design family made no difference on this benchmark, because the look-alike pairs share basis of design and neighbouring zones. Removing clash type, phase and level slightly improved nDCG@5 (0.927), because those fields were the noisiest in both records and queries.

These are results on a synthetic benchmark whose labels follow from how it was generated. They show how each design choice behaves under controlled, realistic noise; they are not evidence of field accuracy or productivity, which require authorized real issue logs and independent human relevance judgments.

6. DISCUSSION

6.1 Implications for practice

The results support a simple design principle: for coordination knowledge, retrieval should be treated as an evidence task rather than an answering task. Three practical consequences follow. First, design-standard revisions should be linked to the issues they affect at the time of the revision: demotion of recorded supersessions was the largest single gain, and unrecorded ones were the main residual error. Second, owners and design teams should capture a few context fields when an issue is resolved; discipline pair and element categories cost seconds to record and carried the most information, while noisy fields such as phase can hurt. Third, any language-model summary should be optional, cited and allowed to say “no support”.

In an industrial portfolio the framework maps onto a cloud coordination platform’s issue interface: a webhook on issue creation triggers retrieval, and reviewer decisions write back as labels. The author has applied an approach of this kind in industrial practice; no data or results from that application are reported here. For new deployments, a shadow-mode pilot is the recommended first step because it yields calibration data without influencing decisions.

6.2 Limitations

Feature matching depends on consistent discipline and element naming; projects that share neither grid naming nor basis of design benefit less, as RQ3 shows. The feature weights and scores were set by judgment and tuned only coarsely; learning them from reviewer decisions is the purpose of the feedback loop. Retrieval cannot recover knowledge that was never written down, so the method works best alongside a lightweight capture step at resolution time. The semantic ranker in the released version is latent semantic analysis; a modern dense embedding model may narrow the gap between text-only and hybrid retrieval, and testing that is a natural next step. Language-model summaries add risk as well as convenience; the framework is useful without them.

6.3 Threats to validity

The generator and labels were written by the author, so the benchmark encodes the author’s view of what makes precedents similar; the look-alike pairs, unrecorded supersessions, noise and query-only vocabulary make it harder, but do not remove that bias. The answer contract was tested with a rule-based simulation, which shows the effect of the contract’s rules but not how well a language model follows them. The test split is small (24 queries with a precedent and 3 without), so intervals are wide and only the larger differences are statistically distinguishable. No user study was conducted, so no claim is made about time savings. Version 1.1 will add independent human relevance judgments, records written by a second author, a dense semantic baseline and a language-model summary layer.

7. CONCLUSION

Coordination clashes recur across repeat building programs, but the knowledge used to resolve them is rarely reused. This article proposed a human-in-the-loop framework that retrieves resolved precedents using engineering context as well as text, demotes precedents made stale by design revisions, constrains any generated summary to cited evidence, and records each engineer’s decision as a label. On an open synthetic benchmark with deliberate hard negatives and realistic noise, the hybrid with demotion raised nDCG@5 from 0.825 to 0.915, cut look-alike precedents in the top five from 9.2% to 2.5% and stale ones from 10.8% to 5.8%, and abstained correctly when no precedent existed. The largest single gain came from recording when a design revision supersedes a resolution. The benchmark, code and run files are released so that other researchers can test retrieval methods for coordination knowledge on shared data, and so that owners can evaluate the approach on their own authorized records.

DATA AVAILABILITY

The benchmark generator, data (records, queries and labels), code, configuration, results and tests are openly available at https://github.com/roja6055-tech/bim-coordination-retrieval [Zenodo DOI to be added on release]. All data are synthetic, and every number in this article can be reproduced with two commands.

DECLARATIONS

Confidentiality. No employer data, prompts, models or knowledge-base content were used. Industrial context is described in general terms.

Competing interests. The author declares no competing interests.

Funding. This research received no external funding.

Use of generative AI. During preparation of this work the author used Claude (Anthropic) to help draft and edit text and figures. The author reviewed and edited all content and takes full responsibility for the publication.

REFERENCES

Carrillo, P., Ruikar, K., & Fuller, P. (2013). When will we learn? Improving lessons learned practice in construction. *International Journal of Project Management, 31*(4), 567–578. https://doi.org/10.1016/j.ijproman.2012.10.005

Cormack, G. V., Clarke, C. L. A., & Buettcher, S. (2009). 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* (pp. 758–759). https://doi.org/10.1145/1571941.1572114

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science, 41*(6), 391–407. https://doi.org/10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9

Hsu, H.-C., Chang, S., Chen, C.-C., & Wu, I.-C. (2020). Knowledge-based system for resolving design clashes in building information models. *Automation in Construction, 110*, 103001. https://doi.org/10.1016/j.autcon.2019.103001

Hu, Y., & Castro-Lacouture, D. (2019). Clash relevance prediction based on machine learning. *Journal of Computing in Civil Engineering, 33*(2), 04018060. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000810

Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems, 20*(4), 422–446. https://doi.org/10.1145/582415.582418

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems, 33*. https://arxiv.org/abs/2005.11401

Malligarjunan, R. (2026). *bim-coordination-retrieval: An open synthetic benchmark for retrieval of BIM coordination precedents* (Version 1.0.0) [Computer software and data set]. https://github.com/roja6055-tech/bim-coordination-retrieval

Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval, 3*(4), 333–389. https://doi.org/10.1561/1500000019

Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., & Gurevych, I. (2021). BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*. https://arxiv.org/abs/2104.08663

Tran, S. V.-T., Yang, J., Hussain, R., Khan, N., Kimito, E. C., Pedro, A., Sotani, M., Lee, U.-K., & Park, C. (2024). Leveraging large language models for enhanced construction safety regulation extraction. *Journal of Information Technology in Construction, 29*, 1026–1038. https://doi.org/10.36680/j.itcon.2024.045

Uhm, M., Kim, J., Ahn, S., Jeong, H., & Kim, H. (2025). Effectiveness of retrieval augmented generation-based large language models for generating construction safety information. *Automation in Construction*, *170*, 105926. https://doi.org/10.1016/j.autcon.2024.105926

Wu, C., Ding, W., Jin, Q., Jiang, J., Jiang, R., Xiao, Q., Liao, L., & Li, X. (2025). Retrieval augmented generation-driven information retrieval and question answering in construction management. *Advanced Engineering Informatics*, *65*, 103158. https://doi.org/10.1016/j.aei.2025.103158

Xia, Z., Zhong, B., Zhang, S., Zhao, T., & Skibniewski, M. (2026). Graph-driven embedding reinforcement and traceable LLM agent for reliable element alignment in construction report generation. *Automation in Construction*. https://doi.org/10.1016/j.autcon.2026.106816

Zou, Y., Kiviniemi, A., & Jones, S. W. (2017). Retrieving similar cases for construction project risk management using natural language processing techniques. *Automation in Construction*, *80*, 66–76. https://doi.org/10.1016/j.autcon.2017.04.003