2026 European Conference on Computing in Construction
Corfu, Greece
July 12-15, 2026

SEMANTIC MAPPING OF REQUEST FOR INFORMATION (RFI) TO BIM ELEMENTS WITH LARGE LANGUAGE MODELS

Limon Paul Joy, Farzaneh Zarei, and Mazdak Nik-Bakht
Department of Building, Civil & Environmental Engineering, Concordia University, Montréal, Québec, Canada

Abstract

Requests for Information (RFIs) remain a critical yet poorly integrated component of construction workflows, typically managed as unstructured text disconnected from Building Information Modeling (BIM) data. Existing studies primarily focus on reducing RFI frequency or response time, with limited focus on linking RFI content to specific BIM elements. This study presents a data-driven framework that combines large language model (LLM)-based intent extraction with BIM metadata reasoning through a staged matching pipeline. By separating semantic cue extraction from rule-based element selection, the framework enhances interpretability and robustness. Validation on experimental and real-world datasets demonstrates reliable element and family-level retrieval.

Introduction

The Request for Information (RFI) is a formal communication procedure to seek information, clarification, or resolution of uncertainties arising in construction. As an essential communication tool, RFI addresses discrepancies, prevents misunderstandings and reduces the risk of delays, claims and cost overruns. Typically, RFIs are initiated by contractors or subcontractors and directed towards designers or consultants to clarify ambiguities in designs, specifications, or constructability issues. The most common reasons for issuing an RFI are unclear specifications, design element contradictions and unforeseen field conditions (Hanna *et al.,* 2012). Despite this significance, RFIs are widely recognized as an inefficient process in project information flow and design coordination. Mohamed *et al.,* (1999) estimated a mean time of 17-person hours to process a single RFI. The cumulative impact of RFI related delays can be substantial at the project level. Total RFI response delays in a project can account for up to 10% of the overall project duration (Papajohn *et al.*, 2018). One of the primary reasons for this inefficiency is the RFI process being largely manual and document centric. Designers are often required to search through multiple sets of drawings, specifications, and revisions to identify the issue raised in an RFI, assess its implications, and issue a response. This fragmented and manual workflow contributes to elongated response times and a lack of traceability between RFIs and the affected building elements.

On the other hand, building information modeling (BIM) has reshaped the construction industry by introducing a digital framework for integrated project management. BIM is a digital representation of the physical and functional characteristics of a facility, which acts as a shared knowledge base for forming reliable decisions throughout its lifecycle, starting from the conceptual stage to its demolition (*NBIMS-USTM V3 – National Institute of Building Sciences*, 2026). Its significance in the construction industry lies within its ability to integrate data across disciplines within a collaborative environment. Furthermore, it provides a visual and spatial context that enhances the understanding of complex design information, enabling stakeholders to interpret the design more effectively than through documentation alone.

Despite their importance in construction practice, the interaction between RFIs and BIM remains limited. While BIM provides structured and spatially rich project information, RFIs are largely text-based and context-poor. Existing research has focused on improving RFIs within BIM-enabled environments, with limited attention to their systematic integration. This creates a noticeable gap in methods for semantically linking RFIs to BIM elements to support faster interpretation, more accurate responses, and improved traceability. Addressing this gap is increasingly relevant with the growth of digital project data and advances in information extraction and semantic analysis.

This research proposes an approach to link RFIs with BIM models to enhance the efficiency and effectiveness of the RFI process. It introduces a data-driven digital framework that utilizes large language models (LLMs) to semantically interpret unstructured RFI text and link it to structured BIM elements, enabling more efficient and context-aware RFI handling. The role of LLM is limited to extracting structural cues for keyword extension to guide the matching, while the core matching is driven by rule-based BIM reasoning. By establishing a direct connection between textual RFI content and BIM-based building elements, the proposed approach seeks to reduce manual effort, shorten response times, and improve information consistency across project stakeholders.

Background review

BIM use cases in RFI

Building Information Modeling (BIM) is one of the most widely adopted digitalization tools for enhancing collaboration in the Request for Information (RFI) process, as well as for identifying design and coordination issues that typically generate a high volume of RFIs. Previous studies have demonstrated that improved coordination within BIM models can significantly reduce the number of RFIs by minimizing design ambiguities and information gaps (Bhat *et al.,* 2017). Similar results were also found by Dan *et al.,* 2023,where the usage of BIM introduced better interdisciplinary design coordination, effective clash detection, and enhanced communication among project stakeholders, leading to reduced RFI frequency. Furthermore, studies investigating the combined implementation of BIM and lean construction principles reported reductions in RFI numbers of up to 70%, highlighting the potential of integrating digital modeling with process optimization strategies (Bhat et al., 2018). Beyond reducing RFI volume, BIM supported workflows also contribute to faster response times, improved design decisions, and more proactive issue resolution, thereby improving overall project efficiency. Morales *et al.,* (2022) identified reactive and proactive uses of BIM application to mitigate common issues raised in RFIs through 3D coordination, specialty design, design review and as-built modelling. BIM uses have also proved to have better financial management regarding RFI related costs. Giel *et al.,* (2013) quantified the return on investment of BIM by comparing BIM-based and non-BIM projects of similar scale, illustrating that BIM enabled projects resulted in reduced RFIs, improved coordination, fewer change orders, and faster response times, leading to significant cost improvement.

However, although substantial research has been conducted on BIM enabled environments and their impacts on the RFI process, the majority of existing studies have primarily focused on three key performance aspects of RFIs: cost, frequency, and response time. These studies have largely evaluated BIM as a coordination and visualization tool to reduce the overall burden of RFIs, rather than examining how RFI itself can be integrated within digital project environments. One of the few studies which focused on the linkage was conducted by Mao *et al.,* 2007 for generating structured RFIs by linking them to structured models such as IFC, enabling easier navigation and retrieval of relevant information for construction professionals. However, despite these early contributions, existing approaches remain largely conceptual and limited in scope, with minimal emphasis on automation. They do not adequately address automated interpretation of unstructured RFI text or its systematic linkage to BIM elements, leaving the process reliant on manual effort. The absence of explicit RFI–BIM connections limits traceability and hinders automated impact assessment.

Integration of LLM within BIM design environment

Recent advances in large language models (LLMs) have enabled new forms of interaction with BIM by enabling semantic interpretation of natural language and facilitating more intuitive interaction with complex model data. This capability has motivated growing research interest in integrating LLMs with BIM to improve accessibility, automation, and decision support. Existing research has predominantly applied LLMs to natural-language BIM querying, conversational model interaction, and parametric design generation ((Zheng *et al.,* 2023; Fernandes *et al.,* 2024; Jang *et al.,* 2024; Ko *et al.,* 2025). While these approaches demonstrate clear benefits in terms of usability, they remain largely confined to design centric use cases and do not address construction communication connections. More recent work has extended LLM applications toward regulatory and compliance-related BIM workflows. LLM-based frameworks have been proposed for automated interpretation of building codes (Yang *et al.,* 2024; Madireddy *et al.,* 2025) and autonomous compliance checking, demonstrating the potential of LLM agents to retrieve BIM data, execute rule checks, and support model revisions through natural-language interaction. Parallel to these developments, prior research has explored the application of natural language processing techniques to analyze RFI documents, focusing on extracting themes, classifying issues, and identifying key entities from unstructured text (Afzal *et al.,* 2023; Afzal *et al.,* 2024). These studies demonstrate the potential of data-driven approaches to analyze RFI datasets through automated topic discovery, issue classification, and entity extraction. In particular, unsupervised topic modelling identified recurring construction issues, while deep learning-based pipelines have been used to classify RFIs by project phase and issue type and extract key entities out of them.

Despite these advances, existing research primarily focuses on design generation, model querying, visualization, and compliance checking. Limited attention has been given to construction communication processes, and in particular, automated linkage of unstructured construction documents and texts to specific BIM elements. This gap limits traceability and automation in construction communication workflows and demonstrates the need for model centric approaches to explicitly connect construction documentation, such as RFIs, with BIM elements.

Methodology

This study proposes a data-driven framework linking unstructured RFIs to corresponding BIM elements, implemented through a Revit-based prototype. The framework integrates three components: (i) extraction of BIM element metadata, (ii) RFI preparation and intent extraction using an LLM, and (iii) a multi-stage matching and scoring pipeline combining deterministic rules with embedding-based semantic similarity to identify the best-matched elements. The LLM is used solely for intent extraction; while matching and final selection are governed by explicit rules and BIM metadata. The staged pipeline was created based on an initial review of the RFI dataset, which indicated its unstructured and variable nature, requiring a stepwise refinement approach. This staged method evaluates high confidence cues first and progressively narrows the candidate through contextual and semantic filtering, with integrated shortlisting to maintain computational efficiency for large BIM models. *Figure 1* depicts the overall architecture of the proposed prototype. Overall, the prototype operates across three environments. BIM element extraction, visualization, and selection are performed within Revit. Structured cues extraction from RFI text is handled by a GPT-based language model. The multistage matching, scoring, and data management are executed in an external Python-based environment that coordinates interactions between the BIM model and the language model. Finally, a human-in-the-loop interface enables users to review and validate the final element selection.

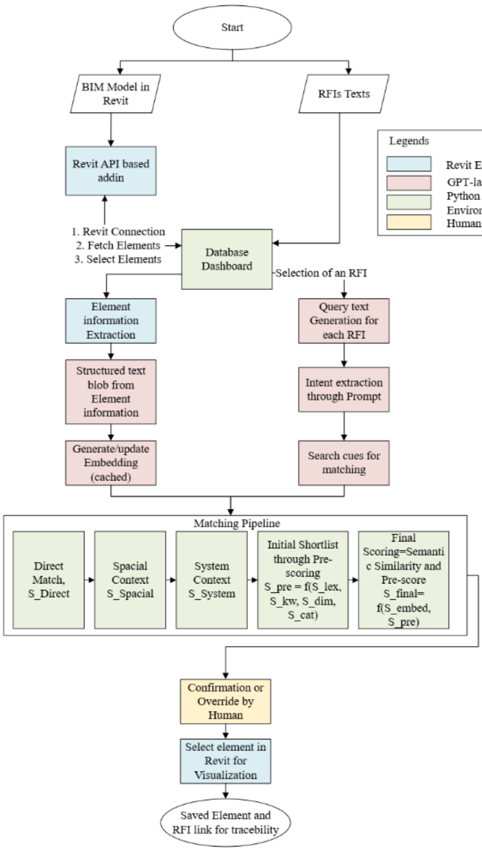

Figure 1: The overall architecture of the proposed prototype

Extraction of BIM element data from Revit model

Autodesk Revit was used as the BIM environment due to its widespread industry adoption. BIM element extraction is a critical step, enabling the transfer of structured geometric and semantic information to external components for semantic analysis and element matching.

BIM model data retrieval

To integrate the BIM model with the LLM-based matching framework, a custom C# Revit add-in was developed using the Revit API. The add-in acts as an HTTP bridge, exposing model data through endpoints for external retrieval, selection, and visualization of elements. This approach was adapted due to Revit API limitations, as API calls must run on the main thread, where computationally intensive tasks such as language processing and embedding generation would otherwise block the interface and degrade performance (*Help | Deployment Options | Autodesk*, 2026; *Help | Revit API Developers Guide | Autodesk*, 2026). The bridge supports three primary operations: Revit connection verifier, element data extraction and selection and highlighting elements.

Element metadata extraction

Using the extraction endpoint, BIM elements are retrieved in JSON format. For each element, a comprehensive set of attributes is collected to reflect how elements are typically referenced in RFIs and coordination documents. *Table 1* includes the parameters extracted from the element metadata.

Table 1: Extracted BIM element metadata parameters.

Category	Extracted attributes
Identifiers	UniqueId, Element Id
Semantic attributes	Category, Family, Instance
Contextual attributes	Level, Room, Space
Tags and Labels	Mark, Type Mark
System attributes	System of MEP elements
Spatial proxies	Nearest grid/axis reference

Structured textual representation and embedding

For text-based comparison, each BIM element's JSON data is converted into a structured textual representation (text blob), organizing key attributes into a normalized format. These text blobs support both (i) lexical/keyword-based matching and (ii) embedding-based similarity computation. Text embedding refers to the transformation of textual data into numerical vector representations that capture semantic meaning. BIM element embeddings were generated using *text-embedding-3-small* and *text-embedding-3-large* for performance comparison and are cached in the database. The embeddings are only recomputed when BIM metadata changes.

RFI data preparation and intent extraction

For semantic mapping of RFI documentation with structured BIM, RFI data is standardized and preprocessed into a consistent textual representation, making it suitable for analysis. Subsequently, structured intent cues are extracted from the normalized RFI text using LLM models.

RFI data preparation

RFIs are imported to the dashboard in a tabular format containing RFI identifier, subject/discipline, question, answer and status. For each RFI, these fields are combined into a single query text to ensure that the contextual information recorded throughout the RFI is incorporated into the matching process. Text normalization is applied to remove accents, standardize casing, and normalize whitespace. This preprocessing of the text mitigates inconsistencies in style, abbreviations and formatting found in real-life practice.

Extract intent cues from RFI through LLM

For extracting relevant semantic cues for BIM element matching from RFI texts, two different large language models (GPT 4.1 and GPT-4.1-mini) were used to extract structural intent cues. Structured intent cues are predefined, machine-readable attributes (e.g., primary object, keywords, location) extracted from RFI text to guide downstream matching and reasoning. The normalized RFI text is fed into the model using a predefined prompt that specifies the exact structure and content of the expected output. The prompt instructs the model to return a strictly formatted JSON object containing only intent-related cues commonly expressed in RFIs, including discipline, primary object description, keywords and synonyms, spatial context (e.g., levels, grid references), and any explicitly mentioned dimensions or quantities. The model is explicitly restricted from inferring BIM element identifiers, Revit categories, or specific element instances. *Figure 2* illustrates the prompt used for intent extraction.

```
You are a construction manager.
Extract structured intent from the RFI text below.
Return ONLY valid JSON with these fields:
- action: one of [move, install, modify, remove, verify, document, clarify, other]
- primary object: a short canonical English noun phrase (e.g., "pendant light", "skylight", "door", "circuit breaker")
- keywords: 5-12 keywords/phrases relevant to matching the correct element (include synonyms; include abbreviations)
- location: location info if mentioned (e.g., level, zone, direction, grid/axis), or empty string
- dimensions: dimensions or quantities if mentioned (e.g., 80 mm), or empty string
- discipline: one of [architectural, structural, mechanical, electrical, plumbing, general, unknown]
- confidence: number between 0 and 1
Rules:
- Do NOT guess BIM element IDs or Revit categories.
- Do NOT suggest specific element instances.
- Return JSON only. No extra text.
RFI TEXT: {RFI_TEXT}
```

Figure 2: Prompt used for intent extraction from RFI text.

In this process, the LLM does not participate in element selection; instead, extracted intent cues guide matching by expanding keywords, activating discipline-specific filters, and identifying spatial or dimensional constraints. Intent influences scoring only when supported by corresponding BIM metadata, and since the LLM cannot introduce new elements or identifiers, hallucinations have no direct impact. This design preserves transparency and limits hallucination effects through rule-based decision-making.

Multistage matching and scoring pipeline

The proposed framework employs a five-stage scoring and matching pipeline to select the best matching candidate. Early stages of this pipeline rely on explicit, high confidence cues while later stages incorporate semantic scoring. This staged design improves interpretability, computational efficiency, and robustness.

Direct identifier matching

The first stage evaluates if the RFI text explicitly mentions BIM identifiers such as ElementId, UniqueId, Mark or Type Mark. These identifiers are matched exactly against the cached BIM metadata. When the match is detected, a direct matching score S_{direct} is assigned.

$$S_{direct} = \begin{cases} 1.00 \; if \; ElementID \; match \\ 0.80 \; if \; UniqueId \; match \\ 0.60 \; if \; Mark \; match \\ 0.55 \; if \; Type \; Mark \; match \\ 0 \; if \; no \; match \end{cases} \quad (1)$$

If multiple identifiers are matched for the same element, their scores are accumulated. Because explicit identifiers provide the highest confidence linkage, if one or more candidates are identified at this stage, the pipeline terminates early and downstream stages are skipped.

Spatial context match

If the RFI contains explicit spatial references such as level, room or grid/axis identifiers, the spatial context matching is used. These references are extracted from the RFI and compared against corresponding BIM element attributes, including levels, associated room or space, and nearest axis/grid references derived from the model. Elements inconsistent with the referenced spatial context are penalized or filtered.

To compute axis/grid distance, each element represented by a point, p (location or centroid) and its distance to axis/grid, g is computed in XY plane as the minimum distance to its segment,

$$d(p, g) = min_{s \in g} \; dist_{XY} \; (p, s) \quad (2)$$

Here, s represents a segment of the axis/grid curve. Based on this, the nearest numeric and alphabetic axis/grid are identified and used for the matching.

Finally, A spatial context matching score, $S_{spatial}$ is computed as,

$$S_{spatial} = \begin{cases} +0.25 \; if \; match \\ -0.25 \; if \; conflicts \\ 0 \; if \; no \; match \end{cases} \quad (3)$$

To preserve recall, if the filter eliminates all the candidates, the unfiltered candidate set is retained.

System context match

For mechanical, electrical, and plumbing (MEP) related RFIs, system identifier provides an important functional constraint. When system related keywords are present in the RFI, alignment between the element system name and the extracted intent is evaluated. A system context matching score, S_{system} is computed as,

$$S_{system} = \begin{cases} +0.15 \; if \; match \\ -0.10 \; if \; system \; is \; missing \\ 0 \; if \; otherwise \end{cases} \quad (4)$$

This stage reduces cross-disciplinary mismatches that may arise from semantic similarity alone.

Prescoring for shortlist

Remaining candidates from the past three stages are ranked based on a pre-scoring process which combines multiple scoring.

First, lexical similarity within the RFI query text, Q and element text blob, E is computed as,

$$S_{lex} = \frac{|Token \; (Q) \cap Token \; (E)|}{|Token \; (Q)|} \quad (5)$$

On the contrary, intent cues derived from RFI texts are grounded against element metadata. The keyword boost score, S_{Kw} is computed as,

$$S_{kw} = \min\left(0.35, 0.05 x Nos\ of\ matched\ keywords\right) \quad (6)$$

Furthermore, a dimensional match is computed if the numeric values mentioned in the RFI is a match the element metadata. The score,

$$S_{dim} = \begin{cases} +0.10\ if\ dimention\ match \\ 0\ if\ no\ match \end{cases} \quad (7)$$

Additionally, sometimes, material names exhibit strong lexical and semantic similarity with RFI text. This induces false positives giving materials as the best match. To avoid this issue and as the scope was to connect physical BIM design elements, material-level entities are explicitly penalized. The score, S_{cat},

$$S_{cat} = \begin{cases} -0.35\ if\ category\ \{\text{Materials, Analytical Surfaces}\} \\ -0.20\ if\ category = Mass \\ 0\ if\ otherwise \end{cases} \quad (8)$$

Finally, an overall prescore is calculated as a weighted sum of all the scores, as S_{pre},

$$S_{pre} = w_1 \times S_{lex} + S_{spacial} + S_{system} + S_{kw} + S_{dim} + S_{cat} \quad (9)$$

Here, the lexical similarity score was assigned a weight w_1 = 0.35 to reflect its role as a supporting signal rather than a dominant factor within the scoring pipeline. Lexical overlap is effective when RFIs explicitly mention element names, families, or parameters, but is sensitive to variations in terminology and phrasing. A moderate weight supports early ranking without overshadowing contextual, spatial, and semantic cues, while only top-ranked candidates are passed to the final stage, improving efficiency and accuracy.

Embedding based semantic similarity

After initial shortlisting through the prescoring, semantic similarity is computed using embedding vectors generated by text-embedding-3-small model. Both the RFI query text, Q and the element text blob, E are embedded into a shared vector space. Semantic similarity, S_{embed}, is computed using cosine similarity,

$$S_{embed} = \frac{V_Q . V_E}{|V_Q| . |V_E|} \quad (10)$$

Final ranking and user validation

The final matching score S_{final} is computed by incorporating embedding similarity with lexical similarity as well as remaining evidence-based scorings, while avoiding double counting of the lexical score. Here, a weight is used to prioritize the semantic score instead of the lexical score, as in some instances, the lexical score can overpower due to keyword matches in the element data. In this study, the weight w_2, was counted as 0.75, which can be increased or decreased depending on the user's requirement in the prototype dashboard to put more importance on either the semantic score or the lexical score. However, the weights and scorings were empirically chosen and no sensitivity analysis was conducted.

$$S_{final} = w_2 \times S_{embed} + (1 - w_2) \times S_{lex} + (S_{pre} - (1 - w_2) x\, S_{lex}) \quad (11)$$

Through the candidate final scoring, candidate elements are listed accordingly in an interactive interface. The next step includes user validation, where the user may accept the top-ranked element or select the appropriate one from the candidate list as the best. This step includes a human-in-the-loop validation mechanism that allows the user to review and confirm the prototype's suggestion. Upon confirmation, the selected element is selected in Revit and the confirmed RFI element linkage, along with all score components, is stored to support evaluation and traceability.

Evaluation of RFI-to-BIM Mapping

To determine the feasibility of this prototype, two BIM environments were considered: (i) an experimental Revit architectural Model with manually created RFIs based on the design, (ii) A real project already being completed with a set of real project RFIs.

Experimental BIM model with generated RFIs

For the first environment, an experimental architectural model was used to assess the baseline performance. Ten RFIs were manually created based on the model content to represent typical RFIs. These RFIs intentionally exclude explicit element identifiers (e.g., ElementId or UniqueId) and rely on descriptive cues commonly found in practice, such as element type, approximate location *(e.g.,* west-side skylight*),* or dimensional information *(e.g.,* timber column 228x502 mm*).* This setup ensured matching based on semantic and contextual interpretation rather than direct identifier matching. As the RFIs were created directly from the known model content, a ground-truth correspondence between RFIs and BIM elements could be explicitly defined. Figure 3 illustrates the experimental model, prototype dashboard with intents extracted from the RFI, best results found through the matching pipeline and the selection in Revit through the prototype environment. For quantitative evaluation of the performance, successful linkage was defined using two criteria: 1. Exact element match, where the top-ranked candidate corresponds to the intended BIM element, and 2. Partial match where the correct family or category is identified despite incorrect instance level selection.

The results demonstrate that the prototype successfully identified and ranked relevant BIM elements for the majority of RFIs. Out of 10 RFIs, 8 were successfully linked to the correct BIM elements, while 2 were unsuccessful. In one failed case, the correct family was identified but not the exact instance, and in the other, no element was matched. Treating exact matches as true positives yields a Top-1 accuracy of 80%, increasing to 90% when partial family or category matches are included, indicating effective narrowing of the candidate search space indicating effective narrowing of the candidate search space. From a retrieval perspective, each RFI was treated as a single query, yielding a precision and recall of 0.80. However, due to the limited sample size these metrics demonstrate feasibility rather than generalizable performance. Analysis of the two unsuccessful cases shows reliance on room-level spatial references, where the absence of instance-level room associations prevented spatial grounding. This highlights the dependency on BIM metadata quality rather than limitations in semantic matching. Overall, the staged pipeline effectively prioritizes lexical and contextual cues, enabling accurate ranking when sufficient metadata is available.

Real project model with issued RFIs

The second environment involved a real construction project with an existing BIM model and a set of real RFIs, issued during the project lifecycle. The project had a total of 524 RFIs in disciplines from architectural, structural, building envelope, electrical, plumbing, mechanical and civil works. Unlike the experimental case, these RFIs exhibited greater linguistic variability, incomplete references, informal phrasing, and cross-disciplinary content, reflecting the complexity of real-world construction communication.

As the disciplines were modeled in separate Revit files, three were selected for implementation: architectural, structural, and building envelope, generating a total of 316 RFIs. In practice, most RFIs were communicated via emails with limited textual detail, with key context provided in separate marked drawing attachments. As this study considers only RFI text, a limited subset contained sufficient semantic information for meaningful linkage to BIM elements. After carefully screening all available RFIs, a subset of 74 RFIs was obtained where the affected design elements could be reasonably inferred. These RFIs were selected based on sufficient textual and contextual information for meaningful linkage. In most cases, identification relied on indirect spatial references (e.g., grid/axis and level information) rather than explicit element identifiers, with only a limited number of RFIs including detailed attributes such as tags, marks, or precise location descriptors that enabled direct mapping. The analysis was conducted under two language model configurations: (i) GPT-4.1 with text-embedding-3-large and (ii) GPT-4.1-mini with text-embedding-3-small, to compare the performance of higher-capacity and lightweight models. Additionally, both configurations were evaluated under two conditions: with intent extraction from RFI text and without intent extraction.

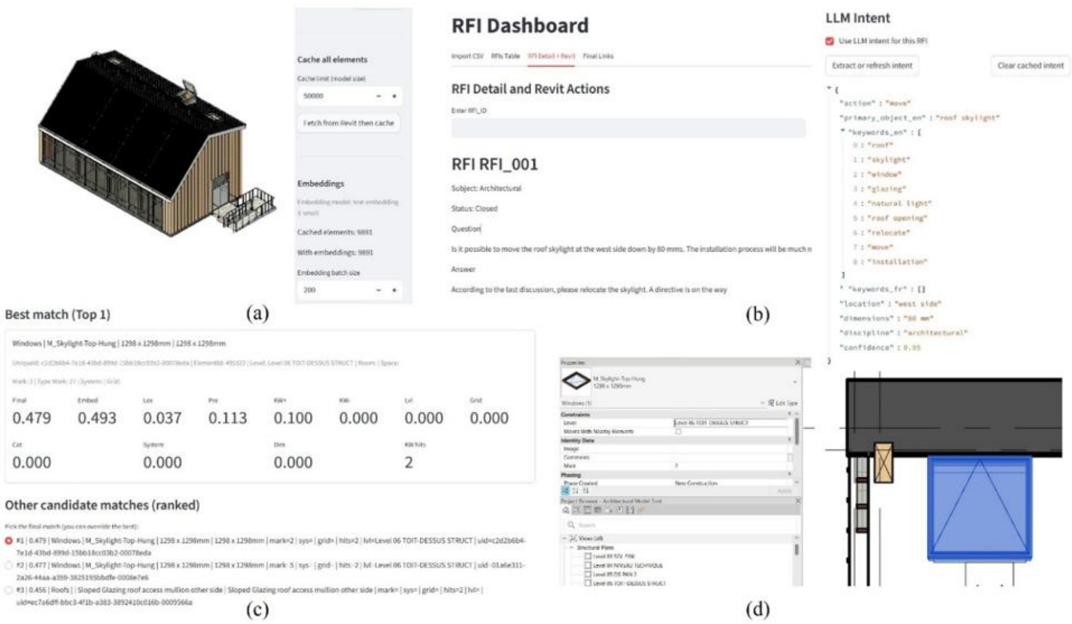

Figure 3: (a) Experimental BIM model, (b) Prototype dashboard with extracted intent from RFI text, (c)Ranked candidate list through the matching pipeline with best match element highlighted separately, (d)Best element selected in Revit

This experimental design enables a systematic evaluation of whether increased model capacity leads to measurable improvements in handling ambiguous and unstructured RFIs.

The results indicate that the framework is particularly effective in narrowing down candidate elements at the category or family level, rather than consistently achieving exact element-level matching. Exact element-level matching was achieved in a limited number of cases, with 4 RFIs correctly linked under configurations with intent extraction and 3 without across both LLM models. These cases typically involved RFIs with explicit identifiers such as element marks, tags, or clear location references. For example, the RFI *"The door supplier from Groupe Brosseau requests clarification on the door pair C-4EX6…?"*, was accurately matched by all models to the element instance due to the presence of specific mark and level information. Although exact matching was limited, the models performed well at category or family-level identification. Of the evaluated RFIs, 19 were correctly predicted by all configurations, 52 by at least one model, and only 1 was not correctly predicted by any model. This demonstrates effective narrowing of the search space by linking RFIs to semantically and functionally similar BIM elements, consistent with real scenarios where RFIs rely on indirect spatial cues. RFIs such as *"Here is a question* *for the architect. What is the window type for R01? According to the elevation and the tab..”* were correctly linked by all models due to matching identifiers in both text and BIM metadata. By contrast, a failure case was observed when both the RFI text and BIM metadata lacked sufficient alignment of detail. For instance, the RFI describing parapet conditions based on multiple drawing references and attachments, the absence of explicit identifiers and insufficient corresponding BIM metadata prevented all models from achieving both element-level and category-level matches. This highlights the dependency of the approach on the availability of consistent textual and model-based information. Comparison across models shows that intent extraction does not significantly improve performance. For GPT-4.1, accuracy remains unchanged at 60.27% with and without intent. By contrast, GPT-4.1-mini shows a slight decrease when intent is used with 56.16%, compared to 58.9% accuracy with intent extraction not being used. This suggests that performance is primarily driven by rule-based matching and BIM metadata rather than LLM-derived cues. Additionally, GPT-4.1 shows more stable performance, while GPT-4.1-mini is slightly sensitive to intent inclusion, indicating greater consistency in higher-capacity models. The results indicate that although exact element-level matching is challenging due to ambiguous RFIs, the framework effectively retrieves relevant candidate elements. The high number of RFIs correctly predicted by at least one configuration demonstrates robustness, while the few unmatched cases indicate strong baseline performance even with limited context. Although the prototype shows promising feasibility by narrowing candidates to relevant categories or families, further validation with larger real-world datasets is required.

Discussion

The study demonstrates that linking unstructured RFI text to BIM elements is feasible under appropriate data and modeling conditions. The prototype contextualizes RFI information within the BIM environment by restricting the LLM to intent extraction and relying on rule-based matching with BIM metadata, ensuring transparency and minimizing hallucinations. Analysis indicates that the primary challenge in automated RFI–BIM linkage is not semantic interpretation, but the misalignment between how RFIs are authored and how BIM models represent design information. In the experimental setting, RFIs with clear element, spatial, or dimensional cues were accurately linked, whereas in the real project, references to repeated elements or design intent led mostly to category- or family-level matches with few exact matches. This observation highlights a key limitation of the current framework, which assumes a primary linkage between an RFI and a single BIM element. While this supports structured evaluation, it does not fully reflect real-world practices, where RFIs often relate to multiple elements or assemblies. Accordingly, extending the framework to support multi-element associations is an important direction for future work. A second major limitation is that many RFIs rely on marked drawing attachments for context, making textual content alone insufficient for interpretation. As the current framework operates primarily on text and does not incorporate such attachments, integrating multimodal data sources, including annotated drawings and visual inputs, is a key direction for future research to enhance contextual understanding and matching accuracy. Another limitation of this study is the dependency on the quality and completeness of BIM metadata. The framework relies in part on BIM metadata for rule-based matching, and although supplemented by semantic and lexical similarity measures, its performance may be affected in cases of incomplete or inconsistent model information. A further limitation is that matching weights were empirically selected without sensitivity analysis, which is identified as a future study.

Overall, the findings indicate that RFI BIM linkage is most effective with explicit and well-defined BIM representations. A key limitation of this study is its reliance on a relatively limited dataset, which may affect generalizability. The results underscore the importance of high-quality BIM data, where consistent categorization, complete parameters, and reliable spatial context enhance RFI grounding. In practice, family- or category-level retrieval helps narrow large models to relevant elements for efficient review. Rather than fully automating instance-level matching, the value lies in enhancing human interpretation through improved traceability and context. Future work will address multi-element linking, integration of drawing or multimodal inputs, and validation on larger, more complex real-world datasets.

Conclusion

This study presents a data-driven framework for semantically linking unstructured RFI text to BIM elements using LLM-based intent extraction and BIM metadata-driven reasoning. In the matching pipeline, the LLM is used for intent extraction, while core element matching and decision-making are governed by rule-based BIM reasoning. By combining intent extraction with a rule-based pipeline, the prototype enables context-aware linkage while limiting AI hallucination. Validation with experimental and real-world data demonstrates feasibility and its potential to improve RFI traceability and spatial understanding in BIM environments. The findings suggest that effective integration of LLMs in BIM workflows requires constraining their role to semantic interpretation, while relying on domain knowledge and BIM reasoning for decision-making. Although limitations related to BIM data completeness, vague RFIs, and single-element linkage remain, they reflect real industry conditions and guide future research. Overall, this work provides a foundational step toward more structured, transparent, and traceable RFI–BIM integration.

Acknowledgments

This research was supported by the NSERC Alliance program under grant ALLRP 585937-23, with additional support from the industry partner, CREO Solutions.

References

Afzal, M. et al. (2024) Improving Request for Information (RFI) Processing in Construction Projects using Natural Language Processing (NLP) Techniques. University of Technology Sydney.

Afzal, M., Wai Wong, J.K. and Fard Fini, A.A. (2023) "Unlocking Insights: Analysing Construction Issues in Request for Information (RFI) Documents with Text Mining and Visualisation," IEEE International Conference on Automation Science and Engineering. IEEE Computer Society. Available at: https://doi.org/10.1109/CASE56687.2023.10260517.

Bhat, A.S., Poirier, E.A. and Staub-French, S. (2017) "Investigating the potential of BIM to address project delivery issues," 6th CSCE-CRC International Construction Specialty Conference 2017 - Held as Part of the Canadian Society for Civil Engineering Annual Conference and General Meeting 2017, 2, pp. 955–964.

Bhat, V., Trivedi, J.S. and Dave, B. (2018) "Improving design coordination with Lean and BIM, an Indian case study," IGLC 2018 - Proceedings of the 26th Annual Conference of the International Group for Lean Construction: Evolving Lean Construction Towards Mature Production Management Across Cultures and Frontiers, pp. 1206–1216. Available at: https://doi.org/10.24928/2018/0479.

Dan, W. et al. (2023) "Potentials of Building Information Modelling (BIM) in Managing Variations," Journal of Advanced Research in Applied Sciences and Engineering Technology, 30(2), pp. 439–456. Available at: https://doi.org/10.37934/araset.32.2.439456.

Fernandes, D. et al. (2024) "A GPT-Powered Assistant for Real-Time Interaction with Building Information Models," Buildings 2024, Vol. 14, Page 2499, 14(8), p. 2499. Available at: https://doi.org/10.3390/buildings14082499.

Giel, B.K. and Issa, R.R.A. (2013) "Return on Investment Analysis of Using Building Information Modeling in Construction," Journal of Computing in Civil Engineering, 27(5). Available at: https://doi.org/10.1061/(ASCE)CP.

Hanna, A.S., Tadt, E.J. and Whited, G.C. (2012) "Request for Information: Benchmarks and Metrics for Major Highway Projects," Journal of Construction Engineering and Management, 138(12), pp. 1347–1352. Available at: https://doi.org/10.1061/(ASCE)CO.1943-7862.0000554.

Help | Deployment Options | Autodesk (2026). Available at: https://help.autodesk.com/view/RVT/2026/ENU/?guid=Revit_API_Revit_API_Developers_Guide_Introduction_Getting_Started_Using_the_Autodesk_Revit_API_Deployment_Options_html (Accessed: March 30, 2026).

Help | Revit API Developers Guide | Autodesk (2026). Available at: https://help.autodesk.com/view/RVT/2026/ENU/?guid=Revit_API_Revit_API_Developers_Guide_html (Accessed: March 30, 2026).

Jang, S. et al. (2024) "Automated detailing of exterior walls using NADIA: Natural-language-based architectural detailing through interaction with AI," Advanced Engineering Informatics, 61(9), p. 102532. Available at: https://doi.org/10.1016/j.aei.2024.102532.

Ko, J., Ajibefun, J. and Yan, W. (2025) "Generative AI-powered parametric modeling and BIM for architectural design and visualization," Proceedings of the Design Society, 5, pp. 1943–1952. Available at: https://doi.org/10.1017/pds.2025.10208.

Madireddy, S. et al. (2025) "Large Language Model-Driven Code Compliance Checking in Building Information Modeling," Electronics 2025, Vol. 14, Page 2146, 14(11), p. 2146. Available at: https://doi.org/10.3390/electronics14112146.

Mao, W., Zhu, Y. and Ahmad, I. (2007) "Applying metadata models to unstructured content of construction documents: A view-based approach," Automation in Construction, 16(2), pp. 242–252. Available at: https://doi.org/10.1016/J.AUTCON.2006.05.005.

Mohamed, S., Tilley, P.A. and Tucker, S.N. (1999) "Quantifying the Time and Cost Associated with the Request for Information (RFI) Process in Construction," Construction Innovation: information, process, management [Preprint].

Morales, F. et al. (2022) "Potential Application of BIM in RFI in Building Projects," Buildings 2022, Vol. 12, Page 145, 12(2), p. 145. Available at: https://doi.org/10.3390/BUILDINGS12020145.

NBIMS-USTM V3 – National Institute of Building Sciences (2026). Available at: https://nibs.org/nbims/v3 (Accessed: March 31, 2026).

Papajohn, D. et al. (2018) "Exploring Potential Delays Associated with Requests for Information in CM/GC Highway Construction," Construction Research Congress 2018: Infrastructure and Facility Management - Selected Papers from the Construction Research Congress 2018, pp. 640–649.

Yang, L. and Zhang, Z. (2024) "Using RAG, LLMs, and langchain to create a query application on open repositories," Proceedings of the ACM/IEEE Joint Conference on Digital Libraries [Preprint]. Available at: https://doi.org/10.1145/3677389.3702488.

Zheng, J. and Fischer, M. (2023) "BIM-GPT: a Prompt-Based Virtual Assistant Framework for BIM Information Retrieval." Available at: http://arxiv.org/abs/2304.09333 (Accessed: February 3, 2026).