Kalpa Publications in Computing
Volume 22, 2025, Pages 589–600
Proceedings of The Sixth International Conference on Civil and Building Engineering Informatics

An Interactive LLM-based Framework for Spatial Relationship Query on Openbim Elements with Ifc4 Schema

Ang Li[1], Peter Kok-Yiu Wong[1], Xingyu Tao[1] and Jack C.P. Cheng[1*]
[1] Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, SAR.
lionel_a.li@connect.ust.hk, cekywong@ust.hk, xtaoab@connect.ust.hk, cejcheng@ust.hk

Abstract

Spatial relationships between BIM elements are crucial for various BIM-based analysis (e.g., identifying external building envelope components for energy simulation), however current BIM information retrieval researches are mainly in the field of extracting attributes from BIM element, a solution for obtaining implicit spatial information from BIM model is needed. Thus, this research focus on how to accurately and effectively obtaining spatial relationship between BIM elements and reliable answering spatial-related BIM query. Addressing this issue, we proposed a IFC-based spatial relation calculation and query-answer framework, which is summarized as follows: (1) extract geometric information of the IFC entities with openBIM standards and obtain the triangulated boundary data of each entities; (2) generate AABB tree of the entities using triangulated boundary data to index the entities and improve the search efficiency in spatial calculation; (3) bSDD-aided LLM workflow to align natural-language queries with corresponding IFC entities to answer spatial relationship queries. We use several building cases to verify our proposed method, the results indicate that our method can accurately understand the natural language query (92.1% correct rate on query understanding tasks) and efficiently determine elements' spatial relationship (saving 61.4% average query time) to answer the original query. With our query method, users with minimal BIM experience (e.g., construction site workers) can still easily query the spatial relationship in a user-friendly way, improving the applicability of the BIM technique.

[*] Corresponding author

I. Hkust, J. Cheng and Y. Yantao (eds.), ICCBEI 2025 (Kalpa Publications in Computing, vol. 22), pp. 589–600

1 Introduction

Identifying the spatial relationship of building elements or building element sets is essential for a wide range of BIM applications across the life-cycle (Tao et al., 2022), e.g., extracting external building envelope components for energy simulation, recognizing the touching elements for laser scanning planning (Chen et al., 2024). The spatial relationship can be referred to topological relationship of spatial elements, which can basically be classified into six type i.e., disjoint, equal, touch, contain, overlap and cover (Borrmann and Rank, 2009). However, the spatial relationship is difficult to directly be obtained from BIM due to the heterogeneous and unstructured BIM data (Wang et al., 2024). The industry foundation classes (IFC) file format, which is the uniform standard for BIM data storage and exchange, didn't define spatial relationship as a general property or attributes of BIM entities. IFC schema provides rich approach to represent geometric and spatial data, for example, swept area representation (IfcExtrudedAreaSolid), boundary representation (IfcFacetedBrep), constructive solid geometry (IfcCsgSolid), each representation approach uses different IFC data structure to record the geometric data (Solihin et al., 2017), but it is recorded with complexity multi-layer attributes inheritance and various attribute formats. Thus, it is valuable to design a framework to automatically extract relevant spatial-geometric information of BIM elements from complex IFC data and calculate the spatial relationship between them.

Usually, a BIM model contains a large number of building elements (e.g., wall, beam column) and various of geometric attribute (e.g., shape, boundary), but they are not indexed based on their spatial position. So, searching an element that satisfy the given relationship (e.g., find out all the touching element of a wall) requires to traverse through all the elements and calculating their spatial relationship (Ying et al., 2022), which is low efficiency since the time cost exponentially increase as the number of elements increase (t_0*2^n, t_0 is the calculation time of intersection test for two elements). Although several spatial indexes (e.g., Octree, Quadtree) to decompose the whole BIM space and improve the efficiency (Borrmann and Rank, 2009), but an BIM object might be divided into two sub-space which may result to computing error. Thus, a powerful spatial index could accelerate the calculation and reduce computational cost is important for the spatial relationship query.

In addition, most of BIM query languages are structured query language-based, users need to learn query grammar and query function to access to the required data or information (Wang et al., 2022). However, not only the BIM professionals will use the BIM query, people who are unfamiliar with BIM also need to query information from BIM, like construction worker need to query about the equipment contained in the room for better installation. Traditional BIM query method might not be user-friendly for all the user. The emerging large language model (LLM) like chatGPT shows excellent performance on context understanding and extracting specific information (Zheng and Fischer, 2023), which imply the potential on semantic processing of spatial query in natural language and interacting with spatial information. Therefore, a methodology is possible to engage LLM to facilitate spatial query to enhance its applicability, so that user even without any background on IFC or SQL can also finish the query.

Against this contextual backdrop, this paper intends to proposes an interactive query method for spatial relationship. With such query method, users can use natural language to query the desired spatial information and effectively obtain the correct answer of the original query, which can promote the application of BIM for unexperienced users.

2 Related Works

2.1 BIM Spatial Relationship

The spatial relationship of BIM elements is a type of implicitly semantic information in BIM models, inferred from the geometric data. In engineering practice, the BIM data including geometric data (e.g., object shape) and non-geometric data (e.g., materials) (Wang et al., 2022), are usually stored in stakeholder-neutral IFC files for the sake of effective information exchange. To mine the spatial relationship in BIM, current approaches usually comprise two steps: (1) Extract geometric data from BIM/IFC file; (2) Determine the spatial relationship using computational geometry (CG) algorithms.

Among all geometric representation, the boundary representation (B-rep) is the most popular one for analysing topological and spatial relationship using CG algorithms (Solihin et al., 2017). The mathematical model for such geometric representations is robust since it can be validated rather easily and consistently, due to obeying the Euler operators. With the normalized B-rep (e.g., triangulated B-rep, using minimal triangles to represent the shape boundary), other B-rep like Axis-aligned bounding box (AABB), Oriented bounding box (OBB) can also be easily obtained (Ying and Lee, 2020). Thus, we use triangulated B-rep to represent the extracted geometric information for facilitating further calculating. Additionally, buildingSMART proposed Model View Design (MVD) to standardize the information exchange requirements (e.g., for energy simulation or quantity take-off (Jeon et al., 2021)), but few MVDs is designed for spatial relationship determination, so we develop an MVD to standardize the necessary geometric information unit and support information extraction algorithm.

The second step for spatial relationships is determining spatial relationships with CG algorithms. Borrmann and Rank (2009) examined the use of the 9-IM as a formal basis for defining topological operators in an BIM environment, which depicts 8 typical topological relationships (i.e., Disjoint, Inside, Equal, Touching, Containing, Overlapping, Covering and CoveredBy) using the 9-IM matrix. The 9-IM approach can cover most of practical spatial relationship, thus our proposed method use it as theoretical basis to determine the spatial relationship.

2.2 BIM query language

Previous BIM query languages (e.g., BIMQL, PMQL) have been developed for filtering objects according to their attribute and type values, manipulating the result set like union, complementary, and get the value of the attributes (Daum et al., 2014). However, none of the BIM query take the geometric representation of IFC objects for the filtering of data into account, so it can't directly execute spatial query or provide information for spatial calculation. Thus, our research integrates the geometric information retrieval and spatial relationship determination into spatial query method to enable the reliable answer can be obtained.

To make the query system more acceptable for nonexpert user, Natural language processing (NLP) is used to enable query system to understand query in natural human language and build a natural language–based virtual assistant to structuralize the original query and support BIM activities (Wang et al., 2022). However, traditional NLP methods (e.g., Word2Vec, BERT) have restricted performance on semantic similarity comparison (Devlin et al., 2019), for example, the "measured net distance" and "NetPerimeter" are different concept (distance and perimeter are disjoint) but are likely to be assigned a high similarity score. The LLM like chatGPT has excellent semantic and context understanding performance (Zheng and Fischer, 2023), but it hasn't been applied to BIM query. Thus, we develop an LLM workflow to understand the NL spatial query and convert it into structured command, which is subsequently utilized to retrieve relevant information and answer the original query.

3 Research Method

Addressing key challenges (complex data extraction, low search efficiency, user-unfriendly interaction) in spatial query of BIM, we use openBIM MVD standard to support IFC data extraction, generate spatial index of BIM elements to improve efficiency, develop LLM-based workflow to understand natural language query of user and answer the query with determined spatial relationship.

As Figure 1 shows, the proposed methodology includes three parts as follows: 1) extract IFC information. In this part, the geometric information contained in IFC files is extracted and the triangulated B-rep are utilized to represent corresponding geometry. 2) generate AABB tree. In this part, the AABB of IFC entities are calculated and the AABB tree structure is engaged to index the IFC entities and improve the efficiency in calculation. 3) understand and answer query. In this part, the keyword in natural language query is identified and align to its corresponding IFC entities to structuralized the query, then relevant information is retrieved to determine relationship, subsequently answer the query in natural language.

Styles for the Article Body

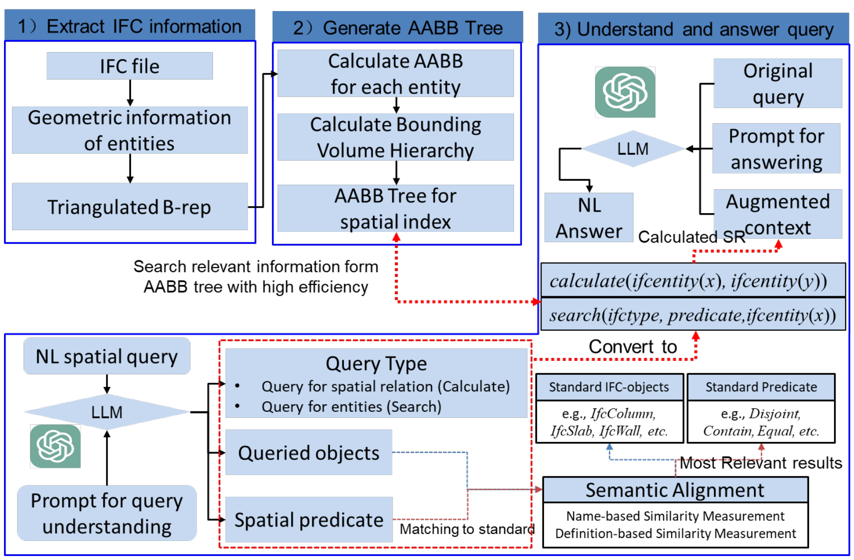

Figure 1: Overall Methodology of the spatial query system

***Part 1*: Extract IFC information.** Extract geometric information of the IFC entities based on openBIM IDM-MVD method. The objective of this part is to convert implicit geometric data (in IFC) into explicit geometric representation. The Model View Definition (MVD) is utilized to standardize the necessary information for determining spatial relationship, where 4 spatial containments (Site, Building, Building Storey, Space) and 6 building elements (Beam, Column, Door, Slab, Wall, Window) as well as their corresponding IFC geometric data structures are defined in the MVD concepts for IFC date reduction and facilitate the spatial calculation. The defined MVD is presented in Figure 2. Besides, the Information Delivery Manual (IDM) can be utilized to extend the MVD to specify the information exchange requirement between different BIM users across the life-cycle. Under the support of these openBIM standards, we use IfcOpenShell to extract relevant geometric information of IFC objects and uniform their coordinate systems, then the obtained spatial data are imported to FreeCAD to generate explicit geometric representation and calculate triangulated B-rep, which can normalize further calculation process for determining spatial relationship (i.e., triangle intersection algorithms) and guarantee computational accuracy.

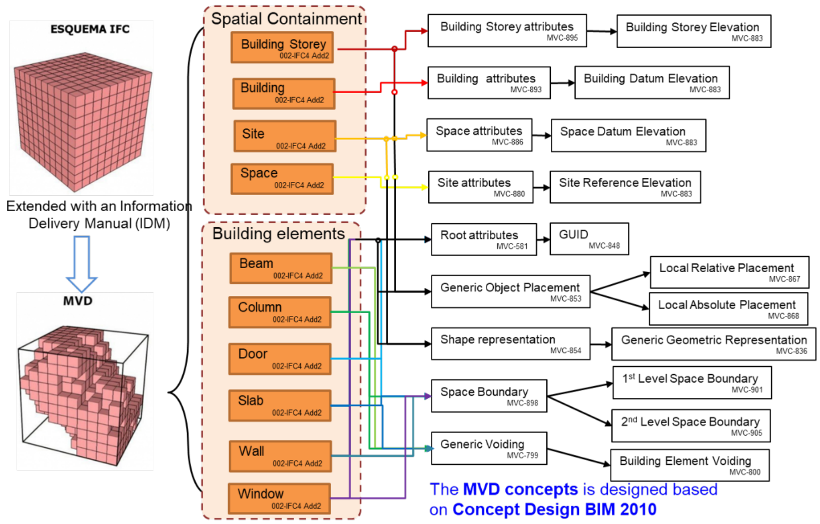

Figure 2: MVD for the spatial relationship

Part 2: **Generate AABB tree**. Spatial index of the entities is generated using triangulated boundary data to index the entities and improve the search efficiency in spatial calculation. Although the triangulated B-rep can be directly utilized to determine the spatial relationship of entities, the numerous objects in large-scale building models may cause the processing time become extremely high. In this step, the AABB tree indexing algorithm is established to index objects in the entire set of extracted IFC entities and accelerate the search efficiency by removing irrelevant candidates, as presented in Figure 3. An AABB tree is a space partitioning tree that indexes geometric primitives (e.g., points, triangles, and polyhedrons) based on their AABBs with $O(nlog(n))$ complexity (better than $O(n^2)$ complexity of without any index), which can be utilized to organizes all the extracted building objects into tree structure using a top-down recursive method.

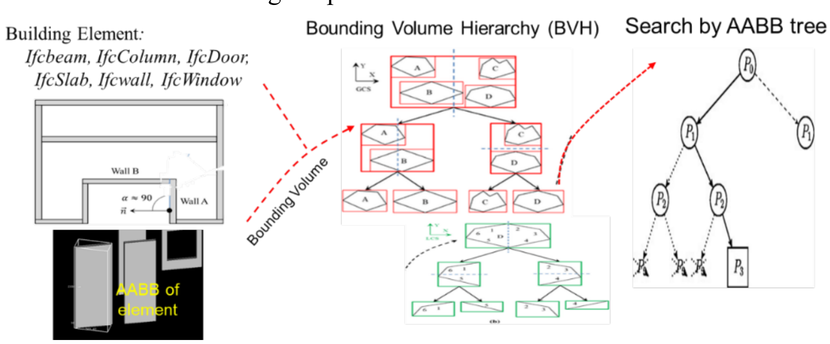

Figure 3: Spatial index by AABB tree

Part 3: **Understand and answer query**. The objective of this step is understanding the natural language spatial query and answering the query with relevant information to make the BIM query more user-friendly. In this step, we developed a prompt pattern for LLM (based on GPT-4) to understand query and translate it into a spatial query function command that can execute by the computer. The prompt pattern includes five components (i.e., system, input query, rules, instruction, output format) as figure 4 shows, which can help the LLM classify query type (i.e., query the relationship of two entities or query the entities with given relationships), identify the query keywords (i.e., relevant building element and spatial predicate), align the identified keywords to IFC entities (e.g., IfcBeam, IfcDoor) and standard spatial predicate (e.g., disjoint, overlap, touch), so that the LLM GPT can understand the original query. The buildingSMART bSDD is utilized to design the rules component, where the name and definition of the IFC entities and spatial predicate provide to help LLM GPT to align the keyword and avoid mis-matching. The original spatial query can be converted into command format by LLM GPT, i.e., *calculate*(*ifcentity*(*x*), *ifcentity*(*y*)) for query the relationship of two entities and *search*(*ifcentity, predicate, ifcentity*(*x*)) for query the entities with given relationships.

Then, the query command is executed to retrieve the relevant geometric information from efficient spatial index, then the spatial relationship of objects is determined using spatial calculation (i.e., triangle intersection test, 9-Intersection Model) according to their geometric information. With the obtained spatial relationship as augmented context in answer prompt, the LLM generate reliable and easy understanding answer for different parties.

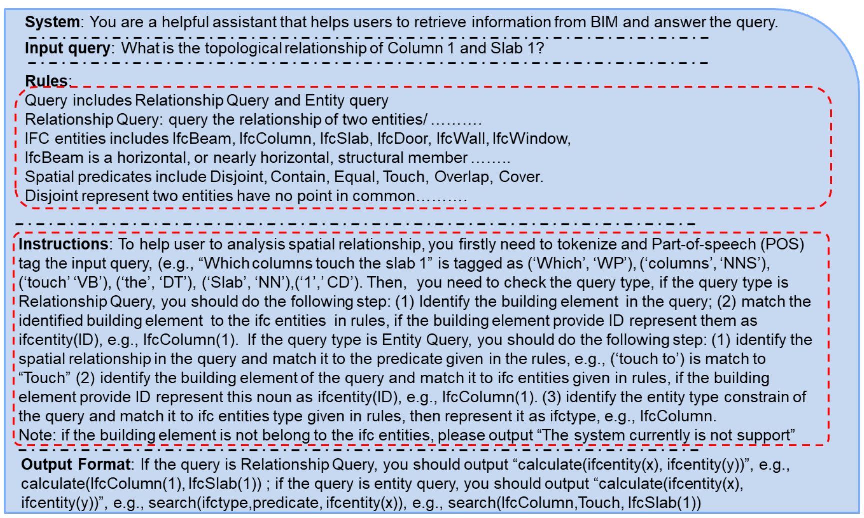

Figure 4: Prompt pattern incorporating bSDD

4 Implementation and Results

We use three practical BIM models (IFC files) in Hong Kong to verify our proposed method as show in Figure 5, i.e., the Cheng Yu Tung (CYT) Building, Shaw Auditorium, Domain Mall. Each selected project's BIM model contains a large number of BIM elements (i.e., Cheng Yu Tung Building (contain 7004 IFC entitles), Shaw Auditorium (contain 33959 IFC entitles), Domain Mall (contain 22177 IFC entitles)). The results of the cases validation indicate that our proposed method can accurately extract geometric information from complex IFC data, generate two-level spatial index for efficient search (average 61.4% search time saving), precisely understand (92.1% correct rate on relationship query understanding and 90.3% correct rate on entity query understanding) and answer the natural language query by LLM to support user-friendly query system of BIM spatial relationship.

4.1 IFC-based Geometric Information Extraction

To demonstrate whether our algorithms can correctly extract information from IFC file and index the elements, we apply our proposed method to the selected cases as Figure 5 shown. The extracting algorithm is developed in Python using IfcOpenShell and FreeCAD toolkit to parse the complex IFC schema and process the B-rep of the entities. The case study indicate that the proposed algorithm can automatically and accurately extract the geometric information of basic IFC entities (e.g., IfcBeam, IfcDoor, IfcColumn) and represent the entities in explicit geometry (i.e., triangulated B-rep) using global coordinate system, which is transformed from the local coordinate system based on IFC schema. In addition, the spatial index of the numerous elements can also be successfully generated based on the AABB tree, which utilize the well-structured tree-based approach to index the unstructured and numerous elements in complex BIM models.

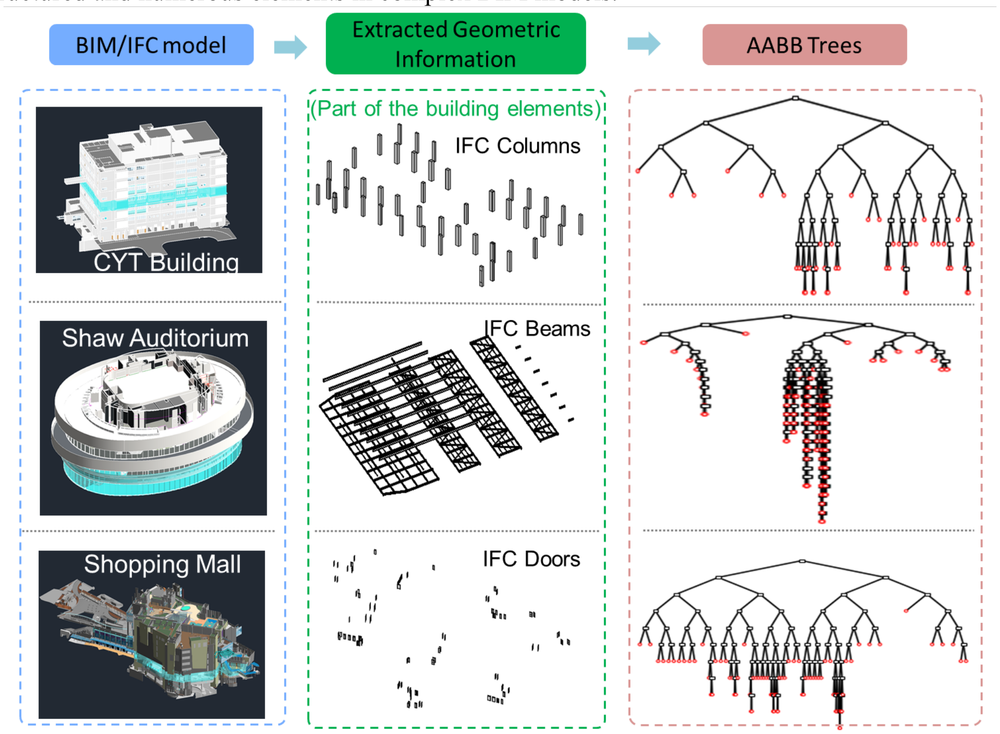

Figure 5: Generating spatial indexes based on geometric information

4.2 Efficiency Improvement of Spatial Index

To verify the efficiency improvement of the spatial index, different index strategies (i.e., the proposed AABB tree and brute force implementation without any spatial indexing) and their average query time on the spatial search tasks were compared, as presented in Table 1. To handle with the high computational cost in traversing through all the relationship of elements, the AABB tree of all the entities is calculated by a top-down recursive method, which use hierarchical AABB to cascadingly partition the space. The results of case study indicate that the proposed AABB tree can significantly improve the search efficiency in different spatial relationship search task (e.g., search touch elements, disjointed elements), the query time is reduced by averagely 61.4% than not using any spatial index. For the BIM model with irregular shape (i.e., Shaw and CYT) and extremely complex layouts, our proposed spatial index still shows excellent performance, which imply the applicability of the AABB tree.

Query Time (s)	Touch			Contain			Cover			Overlap		
	CYT	Shaw	Mall	CYT	Shaw	Mall	CYT	Shaw	Mall	CYT	Shaw	Mall
No index	106.7	346.8	572.2	82.4	302.5	472.5	80.6	358.3	549.0	79.0	371.1	554.2
AABB tree	49.8	131.4	259.6	28.1	98.5	206.8	35.7	116.0	173.2	33.9	119.5	221.2
Improvement	53.3%	62.1%	54.6%	65.9%	67.5%	56.2%	55.7%	67.6%	68.4%	57.1%	67.8%	60.1%

Table 1: Overall query time to obtain spatial relationship by different spatial index

4.3 LLM-based Spatial Query Understanding and Answering

To validate the proposed LLM workflow, we use the actual natural language query to check the output of LLM. Figure 6 (a) shows the original spatial relationship query and Figure 6 (b) shows its output answer of LLM (including the converted command of the original query). The outputs shows that the LLM can understand the given query and convert it to structured command format for spatial calculation. In addition, the LLM can engage the determined spatial relationship form spatial calculation as augmented context to generate easy-understanding natural language answer of the original spatial query, which makes the query process of the spatial relationship more user-friendly.

To evaluate the understanding performance of the LLM on spatial query, we compared the understanding correctness (on query data set) of different LLM prompt patterns. The query data set contains 180 element queries and 180 relationship queries by random combinations of patterns (i.e., relationship query and element search query) and elements (e.g., beam, wall, column) to generate representative spatial queries which can cover different query styles. The difference prompt is designed based on the ablation study, which is presented in Table 2.

The results of ablation study indicate that our proposed prompt has better semantic understanding capability than common prompt (i.e., only give instruction component) in different type of spatial query task since the correctness rate of LLM output for our prompt is 92% while that for common prompt is about 70%. The understanding performance significantly decrease after removing the definition of IFC entities, implying that the providing bSDD concepts can the enhance semantic understanding capability of LLM.

Prompt	Description
bSDD-aided Prompt **(Our)**	System+ Input Query+ Rules (provide the name and definition of IFC Element for semantic mapping) + Instruction+ Output Format
Ablation Prompt-1	System+ Input Query+ Rules (only provide IFC Element name for name-based similarity measurement) + Instruction+ Output Format
Ablation Prompt-2	System+ Input Query + Instruction+ Output Format
Common Prompt	The prompt may include content of original query, the instruction and the format, but it is not clearly divided into different components, the task are directly given to LLM. No example shows to LLM.

Table 2: Summary of different prompt strategies

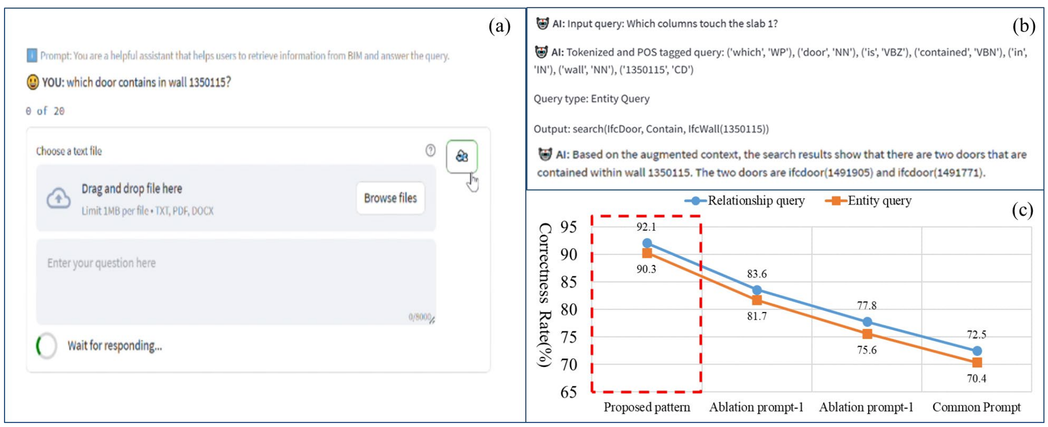

Figure 6: Understand & answer query: (a) Input query, (b) LLM outputs, (c) Performance of different prompt

5 Discussions

5.1 Contributions

This research developed an algorithm for extracting geometric information of building elements (IFC entities) based on IFC schema, a spatial index for generating AABB tree of IFC entities to improve spatial calculation efficiency, an LLM prompt-based workflow to understand spatial query in natural language and answer the query with obtained spatial relationship as augmented context. The contributions of this research can be summarized as follows:

(1) Developed an algorithm for extracting geometric information of building elements based on MVD standard. The buildingSMART MVD can standardize the necessary geometric information for determining spatial relationship, where 4 fundamental spatial containments and 6 basic building elements as well as their IFC data structure are included in the MVD concepts. Under the requirement of MVD, we can develop a Python script to achieve automatic and standardized IFC information extraction, where the IfcOpenShell is utilized to parse the IFC schema and extract geometric

information in python script and the FreeCAD is incorporated to such python script to process the extracted geometric information.

(2) Developed a spatial indexing for BIM elements based on their AABB to improve the search efficiency in spatial query. The AABB tree can convert the massive unstructured building elements into a well-structured tree structure based on the elements' geometric information (i.e., the triangulated B-rep). The proposed indexing system enables search function to rapidly find candidates and remove irrelevant elements by pruning subtree/node, which can the significantly improve search efficiency in spatial query (save 61.4% average query time) and avoid exponential runtime complexity.

(3) Proposed an LLM workflow based on bSDD-aided prompt pattern to understand natural language spatial query and answer the query with relevant information. The proposed prompt pattern for understanding including System, Input Query, Rules, Instruction and Output Format components, where the bSDD concept can be added into rule component to provide name and definition of IFC entities for LLM. Such prompt pattern has shown 92.1% correct rate on relationship query understanding and 90.3% correct rate on entity query understanding, which has 20% higher correctness rate than common prompt. With the understanding prompt pattern, the LLM can identify the keyword in natural language query and align them to standard IFC entity, so that the LLM can convert the original query into command format to retrieve relevant geometric information of the queried object from spatial dataset. The relevant geometric information is subsequently retrieved and utilized to determine spatial relationship and added to answer prompt as augmented context to generate reliable and easy-understanding answer, which make the whole query process more user-friendly than traditional SQL-based query.

5.2 Limitations

While this research contributes to an efficient and user-friendly BIM spatial query system that can use NL to query the spatial relationship, it has several limitations that we need to acknowledge and address.

1. This paper concentrates on the building elements in structural system (e.g., wall, beam), but the elements in different building system (e.g., MEP system) might not be successfully queried since they have different geometric data structures. Querying spatial relationship between different type of elements is meaningful in engineering practices (e.g., relationship of wall and MEP pipe), so the future research is suggested to consider different types of elements in the building.
2. Another limitation is that we currently consider several general spatial queries pattern and the workflow of determining spatial relationship cannot be completely demonstrated during the interaction with the user. So, the future research could address the complex and combinatorial spatial query to facilitate users work, and provide a more interactive query system to help the user understand the spatial relationship.

6 Conclusions

Addressing the key challenges in spatial query of BIM model (i.e., complex BIM data structure, low-efficient spatial calculation and high threshold query language), This research use MVD to standardize the necessary information for determining spatial relationship of elements, which helps to develop an automatically algorithm for extracting IFC geometric information. To decompose the BIM model and index the elements for higher efficiency, this research engage AABB tree for generating spatial index of building elements based on their geometric information extracted from IFC data, which can convert the unstructured elements into a well-structured tree based structure. To enhance the user-friendliness of the query system, this research use bSDD-aided prompt engineering to develop LLM workflow, so that the LLM can understand the BIM-related concepts and answer the query with relevant information.

Three practical projects in Hong Kong are selected to verify our proposed method, the results of case studies indicate that our proposed framework can automatically obtain explicit geometric representation for spatial calculation, significantly improve the efficiency in search relevant spatial data, accurately understand the query in natural language to support user-friendly interaction. Specifically, for the search efficiency improvement, our developed AABB tree spatial index can save about 61.4% average query time for different element search query task. For the understanding performance, the LLM using the bSDD-aided prompt pattern can achieve 92.1% correct rate on relationship query understanding and 90.3% correct rate on entity query understanding.

With our proposed method, non-expert (e.g., construction workers and end user) can still user-friendly query and get relevant information from complex BIM model. Due to the efficient spatial index, users can quickly query and retrieve BIM data, so different parties can spend less time to get the relevant information and respond to their collaborator with lower waiting time, the collaboration can be smoother, thus increasing work efficiency. To achieve higher applicability on different building system and various users, further research is required to consider various elements and equipment (e.g., plumbing) in the building and to support complex and combinational query pattern

Acknowledgments

The authors would like to thank the Innovation and Technology Fund, HKSAR (No. PRP/009/23LI) for partially supporting this research. The authors would like to thank Foshan HKUST Project (FSUST21-HKUST05C) for providing partial support to this research. The authors declare no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Borrmann, A., & Rank, E. (2009). Topological analysis of 3D building models using a spatial query language. *Advanced Engineering Informatics*, *23*(4), 370-385.

Chen, Z., Wang, H., Chen, K., Song, C., Zhang, X., Wang, B., & Cheng, J. C. (2024). Improved coverage path planning for indoor robots based on BIM and robotic configurations. *Automation in Construction*, *158*, 105160.

Daum, S., & Borrmann, A. (2014). Processing of topological BIM queries using boundary representation based methods. *Advanced Engineering Informatics*, 28(4), 272-286.

Devlin J., Chang, M., Lee, K. & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, Vol. 1, p. 2.

Solihin, W., Eastman, C., & Lee, Y. C. (2017). Multiple representation approach to achieve high-performance spatial queries of 3D BIM data using a relational database. *Automation in Construction*, *81*, 369-388.

Tao, X., Liu, Y., Wong, P. K. Y., Chen, K., Das, M., & Cheng, J. C. (2022). Confidentiality-minded framework for blockchain-based BIM design collaboration. *Automation in Construction*, *136*, 104172.

Wang, N., Issa, R. R., & Anumba, C. J. (2022). NLP-based query-answering system for information extraction from building information models. *Journal of computing in civil engineering*, *36*(3), 04022004.

Wang, Z., Sacks, R., Ouyang, B., Ying, H., & Borrmann, A. (2024). A Framework for Generic Semantic Enrichment of BIM Models. *Journal of Computing in Civil Engineering*, *38*(1), 04023038.

Ying, H., & Lee, S. (2020). Automatic detection of geometric errors in space boundaries of IFC-BIM models using monte carlo ray tracing approach. *Journal of Computing in Civil Engineering*, 34(2), 04019056.

Ying, H., Zhou, H., Degani, A., & Sacks, R. (2022). A two‐stage recursive ray tracing algorithm to automatically identify external building objects in building information models. *Computer‐Aided Civil and Infrastructure Engineering*, 37(8), 991-1009.

Jeon, K., Lee, G., Kang, S., Roh, H., Jung, J., Lee, K., & Baldwin, M. (2021). A relational framework for smart information delivery manual (IDM) specifications. *Advanced Engineering Informatics*, *49*, 101319.

Zheng, J., & Fischer, M. (2023). Dynamic prompt-based virtual assistant framework for BIM information search. *Automation in Construction*, 155, 105067.