From Unstructured Data to Knowledge Graphs: An Application for Compliance Checking Problem

Ankan Karmakar[1], Chintan Patel[1], and Venkata Santosh Kumar Delhi[1]

[1]Department of Civil Engineering, Indian Institute of Technology Bombay, India
ankank@iitb.ac.in, 23m0619@iitb.ac.in, venkatad@iitb.ac.in

Abstract –

The rule requirements of a building code are frequently violated to create financially viable designs. These deviations are subjected to condonation by the municipal commissioner if recognizable hardships are faced. The historical concession applications for similar cases are stored in an unstructured manner, creating a barrier to knowledge transfer. The subjective statements given by applicants are composed of logical structure, language, and embedded knowledge that requires years of experience from the domain expert to decipher. A knowledge graph (KG) representation of the problem can capture concepts and represent them visually, which is easy for novice stakeholders to understand. A Large Language Model (LLM)-based method is used in this study for ontology extraction in the form of concepts and relationships. Also, unstructured input preprocessing and entity disambiguation were performed to evaluate the applicability of KG in this domain. The performance of the proposed method was checked qualitatively in a case study from real-life project examples. The limitations and scopes for improvements were also highlighted. The outcome of this study indicates KG as a potential candidate for knowledge generation from the unstructured archival data of compliance checking. The target audience for this application can be the new architects, reviewers, and programmers working on developing the end-to-end automated compliance checking systems. Finally, applying these Artificial Intelligence (AI)-based knowledge transfer mechanisms can ignite future research on automated concession applications and approvals, laying a path to the digital transformation of the industry.

Keywords –

Code Compliance Checking; Knowledge Graphs; Artificial Intelligence

1 Introduction

The construction industry behaves in a fragmented manner where information is not transferred across a building lifecycle, projects, and stakeholders. The siloed data stored prohibits knowledge transfer even among stakeholders from similar domains. Post-design coordination code compliance checking emerges as an essential step for design information transfer. Communication happens between the liaison architects and the developers with government-authorized personnel at the local urban bodies (ULB). According to the literature, automated code compliance systems comprise four steps, i.e., rule interpretation, input model preparation, rule execution, and checking [1]. With the advent of Building Information Modeling (BIM), several commercial and academic research studies have been performed to verify design information at the compliance-checking stage [2]. However, the rule-based compliance checking process is not the end of the complete permit checking process.

After rule checking, the architects apply to the municipal commissioner (MC) to condone several rules that were violated during the design stage. The hardships faced are stated in the form of subjective statements, cross-referencing intertwined concepts. In the Indian context, these problems are detailed in an unstructured manner through natural human languages. Furthermore, the logical reasoning given for achieving a concession in one of the violations can vary across projects. The new architects applying for new concessions or new reviewers looking at applications are deprived of such enriched data captured in a longitudinal timeframe. Even with the availability of the documents, it seems humanly impossible to summarize the tacit knowledge behind these submissions from thousands of documents. Thus, even with the focus on translating code clauses to machine-readable formats, the other half of the compliance checking process lacks the knowledge transfer aspect. Also, during knowledge transfer, an individual actor's own understanding can influence the successor's method of analyzing a present situation, thus reinforcing the requirement of an artificially developed knowledge base.

The system has to capture the unwritten implicit assumptions and complex causal relationships to capture the knowledge from documents prepared by the domain experts [3]. A graph-based framework was proposed in this study to facilitate the knowledge transfer process for compliance checking. The paper applies a Machine Learning (ML)-based graph generation technique to evaluate the applicability of knowledge graphs. Reports were gathered from real-life projects from the municipal corporation of Mumbai, India. Further, the knowledge graph's limitations and potential application areas in the automated code compliance checking domain are also discussed.

2 Literature Review

The literature review is divided into two sub-sections where the knowledge graph (KG) development and its working principle are initially discussed. The following section highlights KG's state-of-the-art applications in the construction industry.

2.1 Knowledge Graphs

Knowledge extraction methods were first developed to address the data available across the web in the forms of text or HTML. These human-created data are not in the form of machine-readable language, leading to the requirement of a system that can extract information from text. This action of extracting insights and inference from the information is knowledge extraction [4]. The steps for the method involve named entity and relationship recognition, entity disambiguation, and relation linking. These methods of knowledge extraction form syntactical relations but lack contextual relations within the knowledge graphs [5]. Knowledge graphs are used to overcome the limitation of contextual relationships among the entities. KG are representations of unstructured data in the form of graphs. It consists of nodes that store the entities, which are interconnected by embedded links, i.e., relationships. Google was the first to introduce its knowledge cloud in 2012, which converted literals to knowledge [6]. Subsequently, several knowledge graphs were developed, such as Wordnet for Natural Language Processing (NLP), YAGO, and DBnet trained over data from Wikipedia. Dbnet model stored data in the form of a Resource Description Framework (RDF) containing subject, predicate, and object triplets [7].

KGs created from unstructured data were indicated to have three components, i.e., ontology extraction, entity extraction, and relation extraction. Further, it was highlighted that relations are not created initially, as manual tagging is required for further training on the extractor network, essentially making it a supervised approach [8]. As KGs have different data formats, several methods are proposed for embeddings, such as - rule-based [5], semantic-based cross-lingual [9], and more. This idea can also be extended to different input formats like text, images, and videos. Further, text and image KGs were combined for visual queries, and text-based embeddings on image data were illustrated, enhancing the visual understanding of the objects by describing them [10], [11].

A few limitations of knowledge graphs include their inability to find the semantic relations with less accuracy and the incapability to explain the relationships. Hence, several applications, such as financial investments, cannot be run based on it. Another limitation is the data's sparsity, especially when working with domain-specific knowledge graphs. As KG are static in nature, but the links or relationships may change over time, the temporality of data also adds to its inadequacy list [12].

2.2 KGs in the Construction Industry

In construction, data is maintained in a siloed manner. Due to this, there is a lack of integration among the data, and engineers lack insights into the decision-making process. KG provides a knowledge management framework that holistically stores the information [13]. KG has various applications in the construction industry. In the case of pavement engineering, maintaining the data related to pavement materials, quality tests during the pavement construction, monitoring data, and updating it in the maintenance phase were identified to be difficult [14]. KG provides the platform to holistically store data in graph formats, with semantic enrichment, which helps project managers make data-driven decisions. A similar use case of the KG graph is explored in Bridge maintenance, where an ontology-based framework for knowledge creation and KG for knowledge storage is used [15]. It also helps in the project management for interoperability in the project teams and document management. A KG model was used with semantic web technologies to highlight the impact of changing design decisions [16].

KGs can also be used to build domain-specific knowledge graphs to analyze construction safety reports [17]. KGs were also used to automate the process of checking the fire safety drawings in combination with BIM. KGs were able to extract the specific information from the clause and match it with the information from the BIM document for the review process [18]. In the digitization of built assets, i.e., digital twins, various information comes in from BIM, IoT sensors, and legal documents [19]. In this scenario, KGs emerge as a valuable tool for project information management. It is based on the ifcOWL-based ontology, providing more openness to the project data [20].

On the other hand, Modular construction, a widely applied technique in the construction of high-rise buildings, requires a high number of customizations, leading to difficulties in managing the work packages. Researchers use KG to manage the data for the extraction of work packages. It maps products to tasks based on granularity and tasks to work packages; subsequently, relationships are learned using the ML models [21].

Knowledge extraction from the video has also been targeted through KG. Computer Vision extracts the entities from the video input, subsequently updating the KG [22]. Similarly, the problem of extracting knowledge from images is also attempted with the ontology of humans, actions, and objects, and the relationships are extracted using ML and deep learning (DL) techniques [23]. Object detection techniques were also used at the construction site to capture the construction progress and derive insights based on the KG created from highly structured data [24].

KG has been applied to the Industry Foundation Class (IFC) graph structure to extract data from BIM models and transfer the knowledge across projects [25]. The BIM-based variants from KGs were retrieved through case-based reasoning and pattern matching for early-stage designs to aid the architects [26], [27]. The BIM KGs were able to suggest alternative design decisions depending on local code restrictions, the similarity of design requirements, and cost-effectiveness [28], [29]. The interoperability among the different BIM authoring tools was also addressed through semantically enriched BIM knowledge graphs produced over the backbone of the IFC graph structure [30].

KGs were recommended for knowledge transfer from subject matter experts to programmers in the compliance checking domain. Researchers also proposed KG quality assessment parameters to generalize the KG application beyond the variations of building codes [31]. Conversely, in the case of unstructured archival data, the variation exists due to the different ways in which the user speaks. According to research, PDF documents hold the lowest rank in the 5-star linked data matrix [32]. Despite the drawbacks of a significant amount of archival data being PDF documents, it cannot be denied that knowledge graphs can be widely applied. The KGs can solve the complex problems of siloed data due to their adaptability in updating the knowledge and cognitive ability to get the semantic links.

3 Application of KGs over Unstructured Data Used for Compliance Checking

The Architecture, Engineering, and Construction (AEC) industry has a plethora of data stored in different formats across projects. The complexity of information increases as the project traverses through the planning to execution phases. A necessary process playing a pivotal role in this information transfer process is the code compliance check. In the Indian ULBs, the checking process is done through AUTODCR [33], which is an automated rule-checking engine for 2D CAD drawings. The information exchanged and reports generated are stored in PDF format, which loses applicability and automatability, thus restricting the transfer of knowledge generated across projects. Applying knowledge graphs to these data helps understand the hidden concepts in these documents and identifies the crucial concepts that need to be stored as a structured database for future use. Where the ULBs hold a large amount of unstructured data, KGs can be a starting point for discovering the paths to move forward.

3.1 Subjective Inputs for Concession Reports

According to the local code "Development Control and Promotional Regulations" (DCPR 2034) for the Greater Mumbai municipal region of India, architects can apply for a concession to the Municipal Commissioner (MC), even after failing to meet the code requirements. The allowable concessions document lists 35 special cases where the MC can provide leeway to the design deviations from the DCPR 2034 code. However, according to clause 6[b] of DCPR – *"In specific cases where a clearly demonstrable hardship is caused, the Commissioner may, for reasons to be recorded in writing, by special permission permit any of the dimensions prescribed by these Regulations to be modified, ..., provided that the relaxation will not affect the health, safety, fire safety, structural safety, and public safety of the inhabitants of the building and the neighborhood."*

The above requirements demanded by the regulation force the architects to demonstrate the hardship faced through writing. The logical reasonings provided under health, fire, structural, public, and neighborhood safety are thus expressed in natural languages through paragraphs explaining the necessity of concessions to make the project feasible. Depending on the arguments provided in the report, a group of concessions can be approved or rejected for that specific project. Rejection leads to a wastage of time for both designing firms and the ULBs in the form of reworks. Further, the knowledge gathered across these projects is stored as unstructured documents, making it impossible for designers to refer to the archival records for future reference.

3.2 Methodology

The unstructured PDF documents obtained from the ULBs of historical projects are used as input documents. These documents are assumed to consist of concepts or ideas that drive the acceptance or rejection of a concession by the MC. The methodology used aims to extract these concepts and their intertwining relationships. This combination of complex concepts might have been challenging to inculcate for the users across multiple projects.

The first step in the process is to preprocess the archival data. The concession pleadings are sometimes presented in the form of paragraphs or nested tables. The headers of these tables also vary across projects, making the automation process challenging to generalize. Once the tables and nested tables are converted into paragraphs, the document is distributed in chunks for each concession type to maintain its inherent contextual proximity. Contextual proximity is the indicator of concepts originating from the same concession type.

The generated chuck length is ensured to be within the capability limits of the Large Language Model (LLM) to be used. In the LLM model, it is prompted to generate concept and relationship pairs for each chunk. The prompt is engineered to instruct the LLM to extract ontology between key concepts. These key terms may include object, location, roles of person, entity, acronyms, documents, or conditions. However, the names of persons, units of measurement, and acronyms are to be excluded.

Further, it was clarified that the concepts can also have one-to-many relationships. The final output was a JSON object with 'node1', 'node2', and 'edge' parameters. The concept nodes generated are further refined for entity disambiguation. In this study, the concepts were clustered manually. However, this step can also use NLPs, a second pass over LLMs, or graph-based models. In the next step, the contextual proximities and communities are created depending on the relationships' degree of centrality and edge weights through the Girvan-Newman method. Finally, the knowledge graph is generated from the acquired data. The methodology is also depicted in 'Figure 1'.

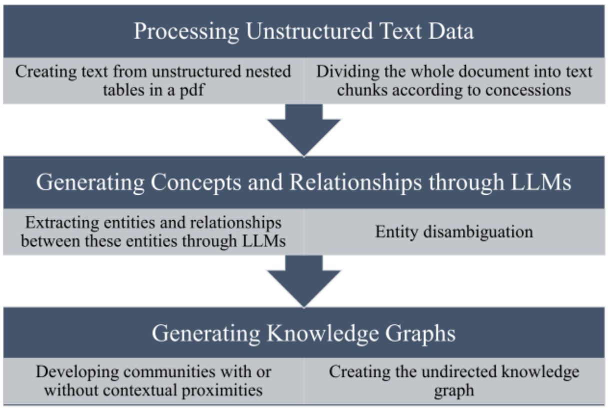

Figure 1. Methodology for developing knowledge graphs from unstructured data

4 Case Study

On submission of designs for pre-construction permits to the ULBs, first rule-based checks on the designs are performed. However, in congested metro cities, many deviations are allowed depending on the planning constraints faced. These concessions are approved through the discretionary power of the MC by reading and understanding the hardships encountered by the architects. PDF reports containing arguments and decisions for these subjecting judgments are stored as archival data in the municipality.

The input data for knowledge graph generation were collected from real-life projects from the Municipal Corporation of Greater Mumbai (MCGM), a ULB from the state of Maharashtra, India. The inputs were concession reports submitted by architects to the Assistant Engineer (AE) of MCGM. Hence, the critical comments from AE were also considered in the analysis. The archival data consisted of submissions that were approved for a set of concessions. Therefore, the entities generated from these reports can be considered as concepts leading to successful condonation.

The LLM used for this study was a quantized open-source model that can be run on a local machine. An open-source model was preferred over a proprietary 'GPT-4' model for trial and error runs. These large models can be used once the basic nodes and relationships are saturated and prompt engineering is finalized. The LLMs used are Mistral-instruct-7b and Zephyr-7b models, where Zephyr generated more profound meaningful entities and relationship triplets than Mistral for this study. Hence, the results discussed in this paper will be output from the Zephyr model.

'Figure 2' depicts KGs generated from the full concession report for two projects with contextual proximities. Meanwhile, 'Figure 3' depicts KGs generated for a single concession case, 'Open Space Deficiency' (OSD). As this graph is based on the conceptual arguments for the condonation of a single concession, contextual proximities are not generated. The final KG' Figure 4' is an amalgamation of concepts generated across ten projects for the OSD concession. The outputs from this graph are further analyzed in the next section.

5 Discussion

The KGs generated from complete reports carry a higher number of relational connections due to contextual proximities. Contextual proximity helps identify the clusters of concepts employed for a specific type of concession. This graph also highlights the potential connections between different clusters. Concession reports are legal documents, so it is a widespread practice to use cross-references to DCPR clauses and previous concession numbers. A central node's high density of sub-nodes also emphasizes the concession type's criticality. Different node clusters and connection patterns can be visualized in 'Figure 2', which points out the variability of the concession reports as an input document for automated code compliance. This finding indicates the complexity of new users /architects going across multiple project types and gathering the logical reasoning required to receive a condonation from the municipal commissioner for any violation in design.

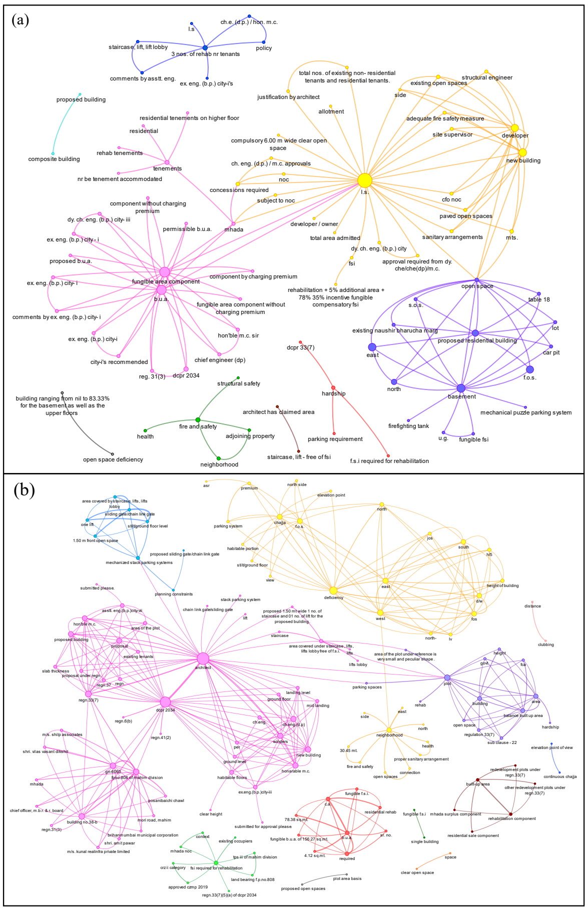

Figure 2. Knowledge Graphs with contextual proximities from two different concession reports

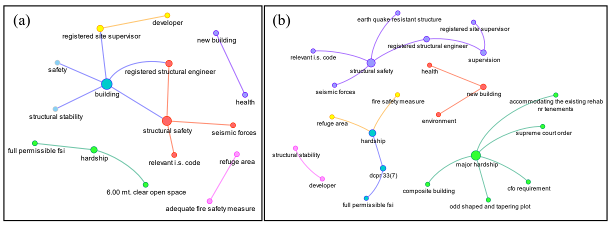

Figure 3. KGs without contextual proximities for OSD Concession for two different projects

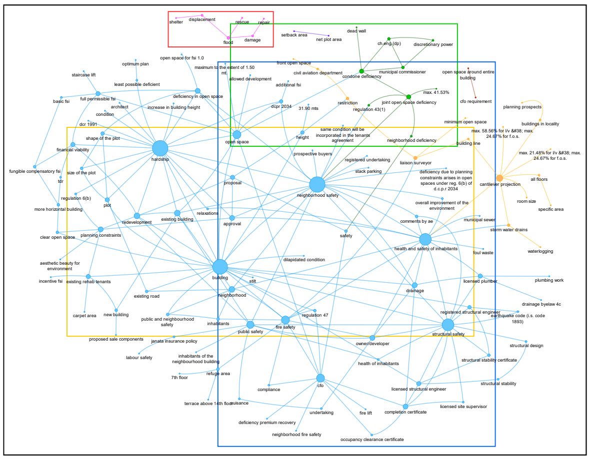

Figure 4. KG for OSD across ten different projects

A specific concession of OSD was considered to identify and analyze the relationship between conceptual entities in detail. In a metropolitan city like Mumbai, with a population density of 73,000 people per square mile, the OSD application was found to be one of the most common concessions across projects. MCGM follows strict guidelines to prohibit the congested development of high-rises. However, after adding allowable fungible floor space index (FSI) and incentive FSI (according to clauses 31(3) and 33(7)) over the basic FSI, the architects are not able to reach the maximum permissible FSI of the building considering the required setback distances. Therefore, the margin requirements are frequently not fulfilled by the design. The OSD application consists of explanations from the architects stating that the margin violation in the required open space will not violate the fire, health, and safety requirements of the inhabitants and the neighborhood. This explanation can vary between projects with a few critical common concepts. The KGs depicted in 'Figure 3' tries to tie these concepts together.

The concepts extracted in part (b) of 'Figure 3' highlight rehabilitation of non-resident tenements, chief fire officer (CFO) requirements, odd plot shapes, placement of refuge area, and consumption of full permissible FSI as significant challenges. The liaison architects also ensured the use of Indian standards for the seismic resistivity of the building alongside the supervision of a registered structural engineer and site supervisor. In the case of project (a), similar hardship concepts were found in a different relationship pattern. The graph generated in project (b) seemed to enlist more concepts than in project (a). However, project (b) seemed to lack connections between the concept nuclei. Due to the variation in the way the architects express, the LLMs failed to create a generalized concept-relationship ontology. OSD concession arguments across ten projects were concatenated to generate a master concept graph, as shown in 'Figure 4', to tackle this challenge.

The KG generated for ten projects confronted challenges regarding concept name ambiguity. For example, different projects used different acronyms to mention the requirement of a no objection certificate (NOC) from the CFO. Hence, entity resolution became crucial for generating the merged concept graph. The likes of entities such as 'cfo', 'c.f.o', 'c.f.o.', 'cfo NOC', 'NOC from cfo', and 'chief fire officer' were clustered under a single entity, 'CFO'. Post entity resolution, the KG developed a complex relationship among different concepts imitating the tacit knowledge, similar to liaison architects. A combination of concepts in the graph can generate an argument to receive condonation of the required OSD successfully. This graph can help architects decide where their design stands in the form of possibilities for concession application.

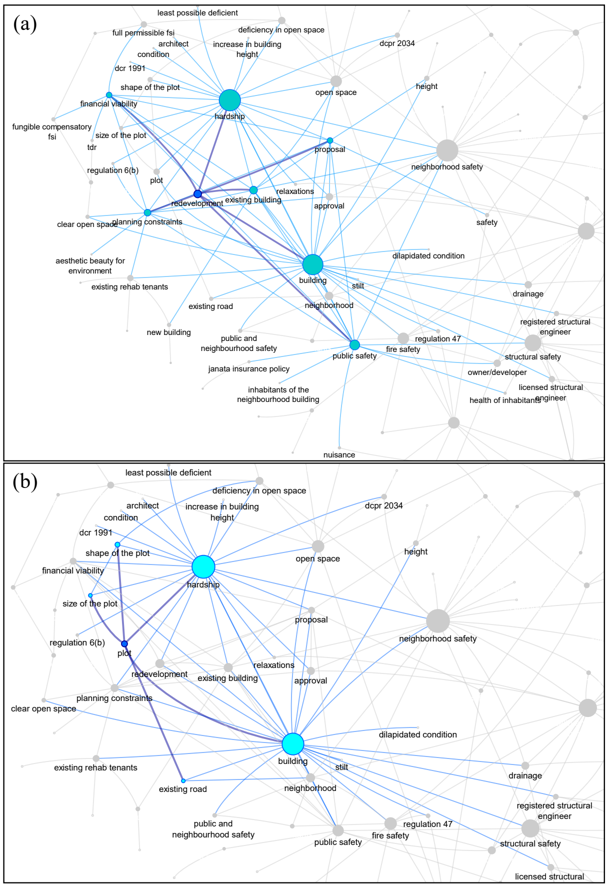

Figure 5. Interaction between concepts leading to hardship in achieving code requirements

The yellow box from 'Figure 4' is zoomed into 'Figure 5' for better visibility. In part (a) of the diagram, an impact of the redevelopment project, whose financial viability is driven by the consumption of fungible compensatory FSI and transferrable development rights (TDR), is found. Furthermore, the interaction between redevelopment projects and public safety was also found. The architects also emphasized planning constraints due to the requirement of clear open spaces and the requirement of existing rehabilitation tenants. Concepts like aesthetic beauty for the environment were an exciting finding, as such hidden concepts might be missed by the human mind while reading reports in the form of PDF documents. In the (b) part, the hardship driven by the plot can be found. On further analysis, the clearance required for existing roads, odd plot shapes, and narrow plot sizes can be identified as the driving factors.

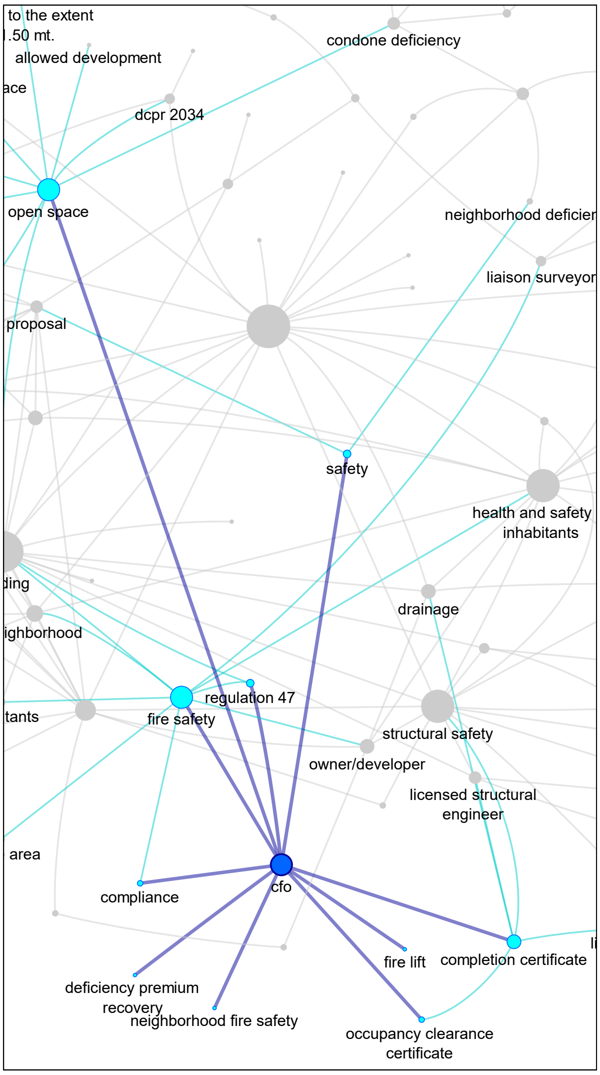

Figure 6. Impact of roles of government personnel in concession acceptance

From the blue box in Figure 4, we derive Figure 6, which shows the role of the CFO in concession approval. With the CFO's NOC, the deficit open space can be condoned, which violates the DCPR 2034 requirements.

Further, it highlights the CFO's decision to influence the completion and building of occupation certifications. Fire safety was also linked with Regulation 47, which indicates the number of fire protection clauses. Thus, the KG performs as a perfect amalgamation for interaction between concepts, government personnel roles, and code requirements.

A similar identification is found in the green box, which is enhanced in 'Figure 7', where the discretionary power of the municipal commissioner and chief engineer is detected. The KG also highlights historical data, such as the maximum joint deficiency approved without charging a premium. The joint deficiency applies to designs with more than one building/wing on a plot, which is correctly connected to neighborhood deficiency, where buildings already exist on the plot.

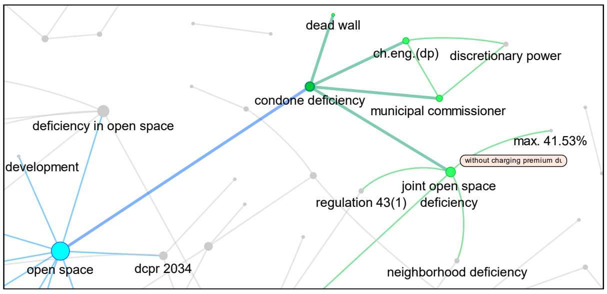

Figure 7. Edge feature and identification of hidden information

A few limitations of the LLM-based entity generation can be identified in 'Figure 8' and 'Figure 9'. In Figure 8, the height restrictions by the civil aviation department near the airport area are correctly identified. However, another entity identified as connected with the height node only summarizes building height, which was also influenced by the Airport Authority of India (AAI) regulations. Even though both these height restrictions indicate the same relationship, the non-standardized way of representing data influenced erroneous concept generation. On the other hand, the entities generated in 'Figure 9' are disconnected from central KG. However, as the concepts identified are highly case-specific, they can get subdued when a large number of project entities are merged, as found in the red box in 'Figure 4'.

6 Conclusion

Deviations and subjective judgments to design hardships in a densely populated metro city are integral to the compliance verification process. The project characteristics leading to successful concession of deviations are often stored as unstructured reports in the ULBs. A method of knowledge transfer for these post-rule-check concession applications is discussed in this paper. Facilitated by the concept graphs, meaningful tacit knowledge was expressed by combining entities and their relationships. The final graph represented complex relationships among concepts, all of which might not be manually extractable without years of domain knowledge. Thus, it can be concluded that, with directed prompt engineering, quantized LLMs can perform noticeably well in generating the node and edge triplets from unstructured data. However, the output suffers from entity disambiguation due to a lack of standardization among report formats and a natural way of expressing arguments. Language models with a significantly higher number of parameters, such as GPT-4, can be used to resolve erroneous results.

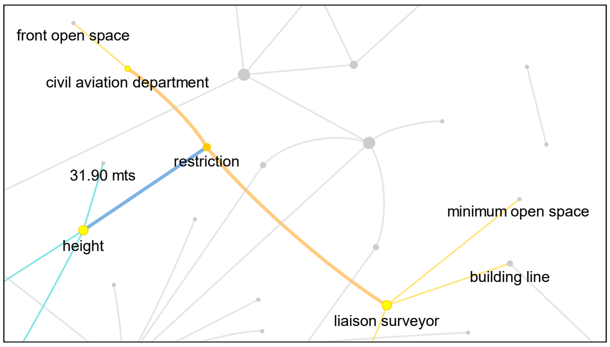

Figure 8. Erroneous entity generation by LLM

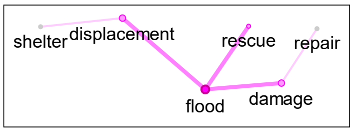

Figure 9. Highly contextualized concepts in KG

The concept graphs generated with the concatenation of knowledge gathered across projects develop a master database. These KGs can be used by new architects, reviewers, or programmers who are likely to develop automated concession application systems in the near future. These systems can gather data for concept nodes identified in the KG from enriched BIM models and predict the acceptability of concession, given the design specifications through ML models. On the other hand, future research can also explore the possibility of generating enriched XML files from KG that can address complex relationships through multiple paths to improve machine readability. Further, the complex reasoning capabilities of KGs can be used across the construction project lifecycle, such as construction progress monitoring and constraint checking. Thus, applying AI-based knowledge transfer mechanisms will only solidify the path for structured data acquisition protocols, leading to the industry's highly required automated compliance checking system.

Acknowledgment

The authors acknowledge and thank for the support received from Softech Engineers Limited through sponsored research project number RD/0120-SOFTT18-001 of IIT Bombay.

References

[1] C. Eastman, J. min Lee, Y. suk Jeong, and J. kook Lee, "Automatic rule-based checking of building designs," *Automation in Construction*, vol. 18, no. 8. Elsevier, pp. 1011–1033, Dec. 01, 2009. doi: 10.1016/j.autcon.2009.07.002.

[2] A. Karmakar and V. S. K. Delhi, "Requirements of Machine Learning and Semantic Enrichment for BIM-Based Automated Code Compliance Checking: A Focus Group Study," in *International Conference on Computing in Civil & Building Engineering*, no. 19, Cape Town, 2023, pp. 65–74. doi: 10.1007/978-3-031-35399-4_6.

[3] W. Solihin and C. Eastman, "Classification of rules for automated BIM rule checking development," *Autom. Constr.*, vol. 53, pp. 69–82, May 2015, doi: 10.1016/j.autcon.2015.03.003.

[4] A. McCallum, "Information Extraction," *Queue*, vol. 3, no. 9, pp. 48–57, Nov. 2005, doi: 10.1145/1105664.1105679.

[5] A. Sakor, "Knowledge Extraction from Unstructured Data," Gottfried Wilhelm Leibniz Universität, 2023. doi: https://doi.org/10.15488/13721.

[6] A. Singhal, "Introducing the Knowledge Graph: things, not strings," *Google Blog*, 2012. https://www.blog.google/products/search/introducing-knowledge-graph-things-not/ (accessed Dec. 19, 2023).

[7] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A Nucleus for a Web of Open Data," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4825 LNCS, Springer, Berlin, Heidelberg, 2007, pp. 722–735. doi: 10.1007/978-3-540-76298-0_52.

[8] R. Reinanda, E. Meij, and M. de Rijke, "Knowledge Graphs: An Information Retrieval Perspective," *Found. Trends® Inf. Retr.*, vol. 14, no. 4, pp. 289–444, 2020, doi: 10.1561/1500000063.

[9] M. Chen, Y. Tian, M. Yang, and C. Zaniolo, "Multilingual Knowledge Graph Embeddings for Cross-lingual Knowledge Alignment," in *IJCAI International Joint Conference on Artificial Intelligence*, Nov. 2016, vol. 0, pp. 1511–1517. doi: https://doi.org/10.48550/arXiv.1611.03954.

[10] Y. Zhu, C. Zhang, C. Ré, and L. Fei-Fei, "Building a Large-scale Multimodal Knowledge Base System for Answering Visual Queries," Jul. 2015, doi: https://arxiv.org/abs/1507.05670v2.

[11] R. Krishna *et al.*, "Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, Feb. 2016, doi: 10.1007/s11263-016-0981-7.

[12] R. Reinanda, E. Meij, and M. de Rijke, "Knowledge Graphs: An Information Retrieval Perspective," *Found. Trends® Inf. Retr.*, vol. 14, no. 4, pp. 289–444, 2020, doi: 10.1561/1500000063.

[13] D. Fensel *et al.*, "Introduction: What Is a Knowledge Graph?," in *Knowledge Graphs*, Cham: Springer International Publishing, 2020, pp. 1–10. doi: 10.1007/978-3-030-37439-6_1.

[14] Z. Yang, Y. Bi, L. Wang, D. Cao, R. Li, and Q. Li, "Development and application of a field knowledge graph and search engine for pavement engineering," *Sci. Rep.*, vol. 12, no. 1, p. 7796, May 2022, doi: 10.1038/s41598-022-11604-y.

[15] Y. Zhang, J. Liu, and K. Hou, "Building a Knowledge Base of Bridge Maintenance Using Knowledge Graph," *Adv. Civ. Eng.*, vol. 2023, pp. 1–16, Apr. 2023, doi: 10.1155/2023/6047489.

[16] M. H. Rasmussen, M. Lefrançois, P. Pauwels, C. A. Hviid, and J. Karlshøj, "Managing interrelated project information in AEC Knowledge Graphs," *Autom. Constr.*, vol. 108, p. 102956, Dec. 2019, doi: 10.1016/j.autcon.2019.102956.

[17] W. Wu, C. Wen, Q. Yuan, Q. Chen, and Y. Cao, "Construction and application of knowledge graph for construction accidents based on deep learning," *Eng. Constr. Archit. Manag.*, no. ahead-of-print, Sep. 2023, doi: 10.1108/ECAM-03-2023-0255.

[18] J. Wang, L. Mu, J. Zhang, X. Zhou, and J. Li, "On Intelligent Fire Drawings Review Based on Building Information Modeling and Knowledge Graph," *Constr. Res. Congr. 2020 Comput. Appl. - Sel. Pap. from Constr. Res. Congr. 2020*, pp. 812–820, 2020, doi:

10.1061/9780784482865.086.

[19] A. Karmakar and V. S. K. Delhi, "Construction 4.0: what we know and where we are headed?," *J. Inf. Technol. Constr.*, vol. 26, no. July, pp. 526–545, 2021, doi: 10.36680/j.itcon.2021.028.

[20] C. Ramonell, R. Chacón, and H. Posada, "Knowledge graph-based data integration system for digital twins of built assets," *Autom. Constr.*, vol. 156, p. 105109, Dec. 2023, doi: 10.1016/J.AUTCON.2023.105109.

[21] X. Li, C. Wu, Z. Yang, Y. Guo, and R. Jiang, "Knowledge graph-enabled adaptive work packaging approach in modular construction," *Knowledge-Based Syst.*, vol. 260, p. 110115, Jan. 2023, doi: 10.1016/J.KNOSYS.2022.110115.

[22] Z. Pan, C. Su, Y. Deng, and J. Cheng, "Video2Entities: A computer vision-based entity extraction framework for updating the architecture, engineering and construction industry knowledge graphs," *Autom. Constr.*, vol. 125, p. 103617, May 2021, doi: 10.1016/J.AUTCON.2021.103617.

[23] Z. Pan, C. Su, Y. Deng, and J. Cheng, "Image2Triplets: A computer vision-based explicit relationship extraction framework for updating construction activity knowledge graphs," *Comput. Ind.*, vol. 137, p. 103610, May 2022, doi: 10.1016/J.COMPIND.2022.103610.

[24] F. Pfitzner, A. Braun, and A. Borrmann, "Object Detection-Based Knowledge Graph Creation: Enabling Insight into Construction Processes," in *Computing in Civil Engineering 2023: Data, Sensing, and Analytics*, 2023, pp. 186–193. doi: 10.1061/9780784485224.023.

[25] F. Heger and D. Napps, "Floorplan2IFC: Transformation of building floor plans into hierarchical IFC-graphs," in *34. ForumBauinformatik*, no. September, 2023, pp. 19–26. doi: 10.13154/294-10084.

[26] D. Napps, D. Pawlowski, and M. König, "BIM-based variant retrieval of building designs using case-based reasoning and pattern matching," in *Proceedings of the International Symposium on Automation and Robotics in Construction*, Nov. 2021, vol. 2021-Novem, no. December, pp. 435–442. doi: 10.22260/ISARC2021/0060.

[27] D. Napps, A. Zahedi, M. König, and F. Petzold, "Visualisation and graph-based storage of customised changes in early design phases," in *Proceedings of the International Symposium on Automation and Robotics in Construction*, 2022, vol. 2022-July, no. July, pp. 191–198. doi: 10.22260/isarc2022/0028.

[28] D. Napps, J. Maibaum, and M. König, "Identifying cost-efficient alternatives to building designs based on building regulations," in *Proc. of the 30th Int. Conference on Intelligent Computing in Engineering (EG-ICE)*, 2023, pp. 1–10.

[29] D. Napps and M. König, "Consideration of detailing in the graph-based retrieval of design variants," *eWork Ebus. Archit. Eng. Constr. - Proc. 14th Eur. Conf. Prod. Process Model. ECPPM 2022*, pp. 389–396, Mar. 2023, doi: 10.1201/9781003354222-50/CONSIDERATION-DETAILING-GRAPH-BASED-RETRIEVAL-DESIGN-VARIANTS-NAPPS-K.

[30] Z. Wang, H. Ying, R. Sacks, and A. Borrmann, "CBIM: A Graph-based Approach to Enhance Interoperability Using Semantic Enrichment," Apr. 2023, Accessed: Feb. 18, 2024. [Online]. Available: https://arxiv.org/abs/2304.11672v1

[31] Y. Zhou, W. Solihin, and J. K. W. Yeoh, "Facilitating Knowledge Transfer during Code Compliance Checking Using Conceptual Graphs," *J. Comput. Civ. Eng.*, vol. 37, no. 5, 2023, doi: 10.1061/jccee5.cpeng-4884.

[32] P. Pauwels, A. Costin, and M. H. Rasmussen, "Knowledge Graphs and Linked Data for the Built Environment," *Struct. Integr.*, vol. 20, pp. 157–183, 2022, doi: 10.1007/978-3-030-82430-3_7/COVER.

[33] SoftTech Engineers Limited, "AutoDCR." https://softtech-engr.com/autodcr/ (accessed Dec. 15, 2023).