Natural Language Query for Power Grid Information Model

Bing Wu[1], Jinhao Cao[2], Yuanbin Song[2,*], Junyi Chu[2], Fulin Li[2] and Sipeng Li[3]

[1]Economic and Technological Research Institute, State Grid Zhejiang Electric Power Co. Ltd., 310001, Hangzhou, China
[2]Department of Transportation Engineering, School of Naval Architecture, Ocean & Civil Engineering, Shanghai Jiao Tong University, 200240, Shanghai, China
[3]Lishui Power Supply Company, State Grid Zhejiang Electric Power Co. Ltd., 323000, Lishui, China
[*]ybsong@sjtu.edu.cn

Abstract. A building information model often provides the functional and physical data of an electrical facility for the downstream construction, operation and maintenance of the built power grid infrastructure. Therefore, rapid and convenient query of the required information from the design model is crucial for all the project participants. However, the query of the design data from a BIM model is frequently burdensome and tedious. Moreover, the Grid Information Modeling (GIM) schema, developed by the China State Grid for describing electrical equipment with more engineering details, exaggerates the difficulty of querying design model. This study applies the Natural Language Interface to Database (NLIDB) approach for querying data from the Neo4j graph database that fuses both IFC data for architectural or structural design and GIM data for electrical equipment. Meanwhile, this study also develops a tool to automatically convert the natural language questions into Cypher queries. In addition, a knowledge graph is also developed for linking the semantic elements extracted from the natural language questions with the IFC semantics stored in the Neo4j database.

Keywords: Power Grid, Computer Aided Design, Building Information Modeling, Natural Language Interface to Database, and Deep Learning

1 Background

Design information is crucial resource for the construction and operation management of an electrical infrastructure, but it has been sealed by a few proprietary CAD formats for a long time until the wide application of the Building Information Modeling (BIM) approach. A building information model is the digital representations of the physical and functional characteristics of an engineering project, and such a model is often represented by the Industry Foundation Class (IFC) format, an open representation schema for sharing design information among trades. Besides the 3D geometric data, an IFC file also contains data of the compositional,

physical, and functional attributes of an electrical facility. Moreover, the China State Grid issued the domestic BIM standards, often called Grid Information Modeling (GIM) [1], to simplify the description of electrical devices or equipment, which exaggerates the difficulty of query of BIM data.

The research of Natural Language Interface to Database (NLIDB) provides a new means to acquire data from BIM design models [2]. NLIDB tools can automatically generate database queries by translating natural language sentences into a structured format. These tools may play an increasingly important role as designers and engineers seek to obtain information from design model databases without the assistance of computer experts with specific domain expertise or knowledge of formal query languages [3]. Natural language query can greatly reduce the time and cost for designers and engineers. Compared with English text, Chinese sentences are more difficult for processing since there is no gap between Chinese words or characters. Therefore, it is often required to segment a Chinese sentence into a sequence of words with the prevailing tools like Jieba [4], LTP2, and CoreNLP [5].

Although Recurrent Neural Network (RNN) is more suitable for evaluating sequence of data than Convolutional Neural Network (CNN) [6], the traditional RNN approach may not capture the semantic relationship between two words with long distance in a sentence, nor can it solve the network training problem of gradient disappearance and the gradient explosion. On the other hand, Long Short-Term Memory (LSTM) network can make up for the shortcomings of the RNN model with the usage of gate units [7]. Furthermore, Bi-LSTM apply both forward and backward processing for capturing the richer context for a word [8]. In addition, many studies implied that pretrained NLP models can be successfully used as the base for specific applications with supplementary corpus. In general, there are two typical approaches for utilizing pretrained NLP models, i.e. feature-based (for example, ELMo) and fine-tuning (for example, BERT [9]).

Meanwhile, the MAPO (Memory Augmented Policy Optimization) model and later its improved version MAPOX were developed to convert natural language into formal query language [10]. Meanwhile, Dong and Lapata utilized supervised learning strategy to resolve the NL2SQL problem, proposing two alternatives, Seq2Seq and Seq2Tree [11]. Li et. al. used a decomposition strategy for joint extraction of multiple relations and entities from design codes [12]. Nevertheless, the approach of converting the extracted semantic relations from engineers' questions into BIM database query scripts should be further studied. Therefore, this study explores the approach of using natural language to query information models of power grid projects.

2 Framework of Natural Language Query

Fig. 1 illustrates the framework of natural language query on a grid information model. The left part of the diagram shows the procedure to a zipped GIM [13] file is converted into a graph database. In detail, the non-ifc files, in the format of China Grid information modeling data standard, are first converted into .ifc files, using the

method elaborated in the succeeding section. And then all .ifc files are imported into a Neo4j graph database using an IFC file parser and a sequence of Cypher codes programmed in C#. Moreover, a knowledge graph is specially developed for bridging the semantic elements extracted from the natural language questions with the IFC terms defined in the GIM database.

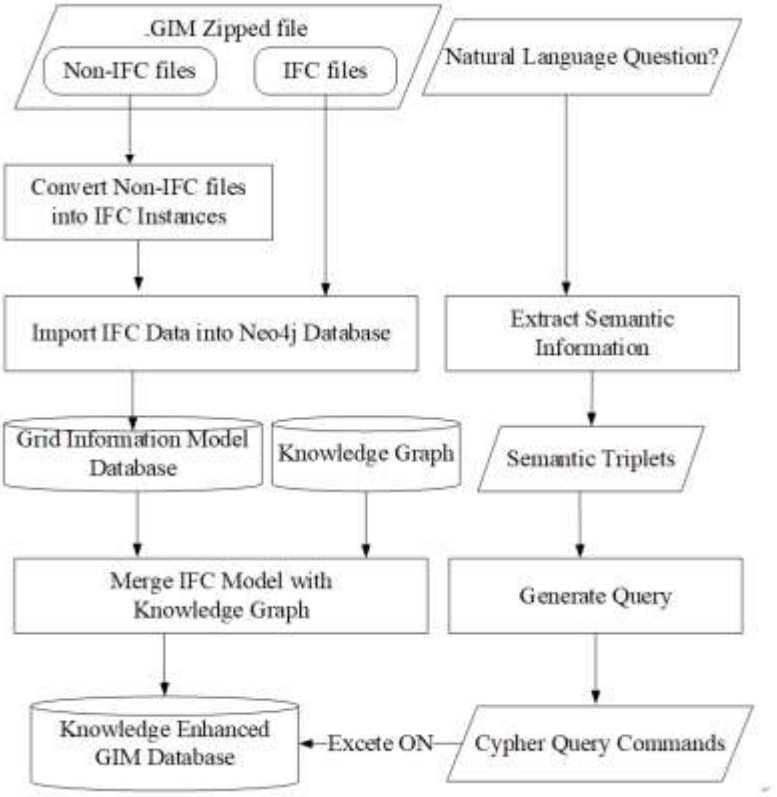

Fig. 1. Framework of natural language query on grid information model

At the same time, the right part of Fig. 1 presents the process flow of for translating a natural language question into a Cypher query script. The semantic information is extracted from a natural language question or query and expressed in the format of triplets. Subsequently, these triplets are automatically converted into Cypher scripts using the templates addressed in the succeeding section. Finally, the Cypher script are executed on a knowledge enhance graph database.

3 Knowledge Enhanced GIM Database

3.1 Conversion from GIM to IFC

A GIM file is actually a zipped file containing 4 folders files: CBM, DEV, PHM and MOD [13]. In each folder, a file uses the GUID as its unique name, and its content is encoded in UTF-8. Specifically, a .mod file depicts the parametric shape with its transformation matrix for local placement and its RGB color. A .phm file describes the structure of multiple .mod files for a more complex component, and one .phm files can further reference other .phm files. The files in the DEV folder define the individual device, equipment or a system of devices. At the same time, the files in the CBM folder describe the organization of subsystems of a gird engineering project, which also comprises .ifc files for architectural, structural, pipe and ventilation systems. In addition, a .fam file is used for depicting the material, physical and functional attributes of either parts or devices or systems.

Since the IFC schema has much richer semantic constructors than GIM, the non-ifc BIM data files for representing electrical devices are first converted into IFC instances. Fig. 2 illustrates a typical case of converting a .mod file into the corresponding IFC components.

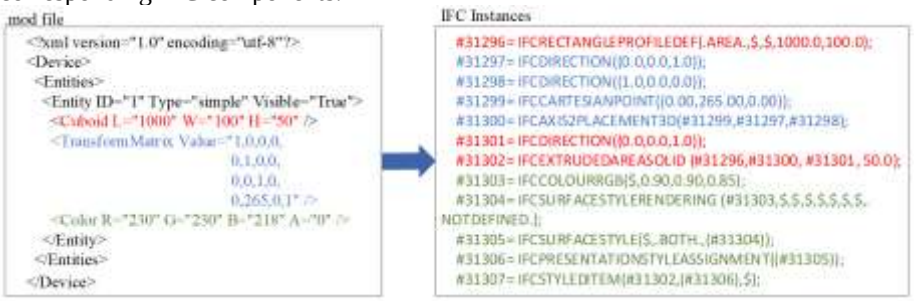

Fig. 2. Conversion of MOD file into IFC instances

The parametric representation of a cuboid in the .mod file is converted a number of IFC instances. The rectangle profile of the cuboid is represented by the IfcRectangleProfileDef instance (#31296), and its extrusion direction by the IfcDirection instance (#31301), and furthermore the cuboid shape is described by the IfcExtrudedAreaSolid instance (#31302). Then, the 4*4 transformation matrix for local placement of the cuboid is also converted into the IfcAxis2Placement3D instance (#31300), which is further described by another three IFC instances, original point (#31299), X direction (#31298) and Z direction (#31297). In addition, the color of the shape is depicted by the IfcColourRGB instance (#31303).

3.2 Knowledge Graph for Enhancing GIM Graph Database

In order to corelate the semantic elements extracted from natural language queries with the class names defined in the IFC schema, a knowledge is defined. The knowledge graph in Fig. 3 illustrates the key concepts: *Element*, *Attribute*, *System*,

Space, Material, Comparison, and MathFunction. Each core concept can be further described with its subcategory concepts or hyponyms.

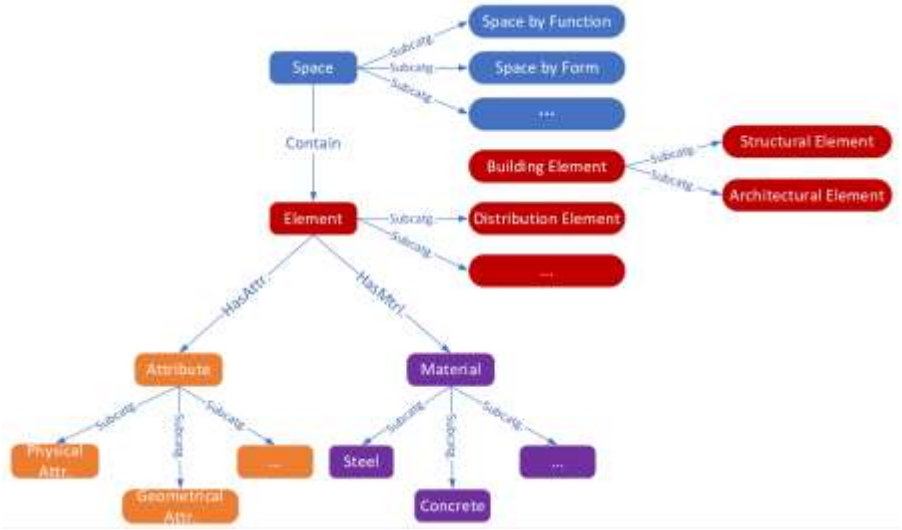

Fig. 3. Part of Knowledge Graph for Enhancing GIM database

In detail, the category of *Element* has 10 sub-categories, i.e. *Building Element, Civil Element, Distribution Element, Feature Element, Furnishing Element, Geographic Element*, etc. Moreover, the *Building Element* can be further divided into *Architectural Element* and *Structural Element*. Subsequently, the category of *Structural Element* can have 6 types of element classes, i.e. *Foundation, Pile, Structural Beam, Structural Column, Structural Slab, and Structural Wall*.

In the knowledge graph, each hypernym associates with its hyponyms by the Subcategory relationship. Meanwhile, there exists HasProperty relationship between Element and Material and Attribute. At the same time, the concept Space has Contain relationship with Element. These semantic relationships between core concepts are also coded as Neo4j relationships.

The key concepts in the knowledge further associate with one or more IFC classes. For example, *Structural Beam connect with* IfcBeam via the ReferenceIfcClass relationship. Then, using the Neo4j Cypher script can create connection between the IFC classes in the knowledge graph and the IFC instances stored in the GIM database. For example, the IfcBeam class in the Knowledge graph can be used for linking the IFC instance labeled IfcBeam in the GIM database. Meanwhile, the Hot Rolled H Steel material concept connected with IfcPropertySet class, which further associates with the IFC instances labeled with IfcPropertySet in the GIM database. In this way, the knowledge graph acts as the bridge between the semantic entities (in the natural language questions) and the IFC instances (in the GIM database).

4 Automatic Generation of Cypher Script

The joint extraction model developed by Li and et. al. [12] is used to extract semantic information from natural language questions Fig. 8 illustrates the information extraction pipeline that composes four components, i.e. character embedding, shared semantic encoder, subject extractor, and object and predicate extractor. Through the pipeline, a query sentence written in natural language can be automatically converted into a number of triplets.

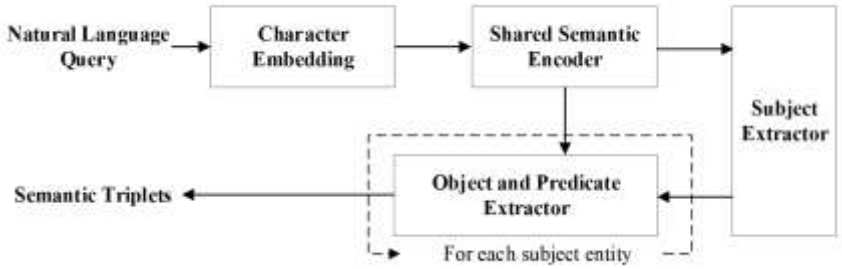

Fig. 4. Framework of the joint extraction model

The character embedding, the first module in the aforementioned pipeline, is utilized for transforming each character in the Chinese question into a real vector, herein called character vector. Subsequently, a shared semantic encoder is applied to learn the context features of each character, literally called task-shared features. Specifically, a Bi-LSTM model is used to encode the association between a character and its surrounding characters (both in its left-hand and right-hand sides). Subsequently, the subject extractor module uses the task-shared features to identify all candidate named entities that have the opportunity to act as subject. And then the associated object entities and predicate relations, for each subject entity identified, are simultaneously identified by the object and predicate extractor using task-shared features. Finally, the semantic information in the natural language question can be automatically extracted into a set of triplets.

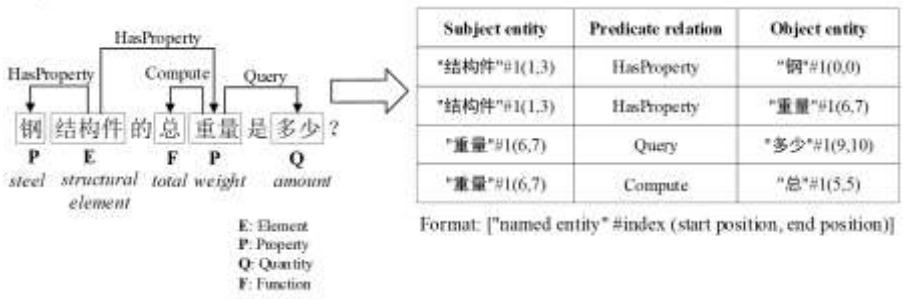

Fig. 5. Conversion of natural language question into semantic triplets

Fig. 5 presents the conversion of the Chinese question “钢结构件的总重量是多少？” to semantic triplets. In English, the Chinese question means to query the total weight of the steel structural elements. Using the pipeline in Fig 4, the natural language is converted into 4 semantic triplets.

```
//(StructureElement, HasProperty, Steel)
MATCH (a:Element)
WHERE a.Name="StructureElement"
MATCH (a)-[*1..5]->(b)
MATCH (b)-[:ReferenceIfcClass]->(c)-[:ReferenceDBIfcInstance]->(d)
MATCH (d)-[:HasProperty]->(e)
MATCH (e)<-[:ReferenceDBIfcInstance]-(f)<-[:ReferenceIfcClass]-(g)
MATCH (g)<-[*1..5]-(h)
WHERE h.Name="Steel"

//(StructureElement, HasProperty, Weight)
MATCH (d)-[:HasProperty]->(i)
WHERE exists(i.Weight)
UNWIND i.Weight as j

//(Quantity, Query, Weight)
RETURN SUM(j) //(Weight, Compute, total)
```

Fig. 6. Conversion templates of semantic triplets into Cypher query

The semantic triplets are then converted into Cypher query scripts via the four predefined mapping templates as shown in Fig. 6. The predict “HasProperty” in the first triplet is translated into Neo4j relationship [:HasProperty] and a set of MATCH commands is simultaneously inferenced to generated linkage path with the assistance of the knowledge graph. In detail, the subject “Structure Element” can be searched from the core concept Element in the knowledge graph, while the object entity “Steel” becomes the property constraint to locate the IFC entities (See the WHERE condition). Meanwhile, the second triple is to locate all IFC element with the property “Weight”, and all the found weight values are organized into a set, named j defined with UNWIND. Finally, the third triplet indicates the goal of the information searching, and the fourth triplet defines the mathematic function SUM of the found weight values, i.e. the total weight.

5 Conclusions

The designers and engineers frequently feel it burdensome and tedious to search information from the prevailing BIM design model of a grid engineering project with multiple query sentences. In this regard, a framework of automatic conversion from natural language questions into Neo4J Cypher scripts has been developed in this study. Those non-IFC files contained in a .gim file are first converted into .ifc files, and then all the .ifc files are imported into the Neo4j graph database to achieve faster information retrieval. And then, the joint extraction model is used for extracting

semantic information from a natural language question into a set of triplets that can be further converted into Cypher scripts by the mapping templates. In addition, a knowledge graph is also developed to connect the semantic entities with the IFC classes. Consequently, the BIM design model can be more effectively and conveniently queried by natural language questions.

Since this research is still in its initial stage, the extraction accuracy of the joint extraction model will be further trained with more labeled questions, and more mapping templates for generating Cypher scripts will also be developed.

References

1. Jung, N., Lee, G.: Automated classification of building information modeling (BIM) case studies by BIM use based on natural language processing (NLP) and unsupervised learning. Advanced Engineering Informatics (41), 100917, (2019).
2. Agrawal, A., Kakde O.: Object-Relational Database Based Category Data Model for Natural Language Interface to Database. International Journal of Artificial Intelligence and Applications 1(2), 35-41, (2011).
3. Sun C.: A natural language interface for querying graph databases. Massachusetts Institute of Technology, Thesis, (2018).
4. Zhang X., Wu P., Cai J., Wang K.: A Contrastive Study of Chinese Text Segmentation Tools in Marketing Notification Texts. Journal of Physics, Conference Series 1302(2), (2010).
5. Manning, C., Surdeanu, M., Bauer, J.: The Stanford CoreNLP Natural Language Processing Toolkit. In: 52nd Annual Meeting of the Association-for-Computational-Linguistics (ACL), Baltimore, MD. 55-60, (2014).
6. Liu, X., Hou, S., Qin, Z., Liu, S., Zhang J.: Relation extraction for coal mine safety information using recurrent neural networks with bidirectional minimal gated unit. EURASIP Journal on Wireless Communications and Networking, 55, (2021).
7. Wang, J., Zhang, L., Chen, Y., Yi, Z.: A New Delay Connection for Long Short-Term Memory Networks. International journal of neural systems 28(6):1750061, (2017).
8. Adnen, M., Mounir, Z.: BLSTM-API: Bi-LSTM Recurrent Neural Network-Based Approach for Arabic Paraphrase Identification, Arabian Journal for Science and Engineering 46, 4163–4174, (2021).
9. Sur, C.: RBN: Enhancement in language attribute prediction using global representation of natural language transfer learning technology like Google BERT. SN Applied Sciences, 22(2), (2020).
10. Liang, C., Norouzi, M., Berant, J.: Memory Augmented Policy Optimization for Program Synthesis and Semantic Parsing. 32nd Conference on Neural Information Processing Systems (NIPS). (2018). Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montreal, Canada December, 10015–10027, (2018).
11. Dong L., Lapata M.: Language to Logical Form with Neural Attention. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 33-43, (2016).
12. Li F., Song Y., Shan Y.: Joint Extraction of Multiple Relations and Entities from Building Code Clauses. Applied Sciences 10(20), (2020).
13. China Electricity Council: Interactive specification for the three-dimensional design model of power transmission and transformation project (2020).