Building and Applying Knowledge Graph in Edge Analytics Environment

Hui Zhang¹, Yingtao Niu¹, Kun Ding¹,†, Shaoqin Kou¹, Liu Liu¹,²

¹The Sixty-Third Research Institute of National University of Defense Technology, Nanjing 210007, China
²School of Information Engineering, Suqian University, Suqian 223800, China
*email: dingkun18@nudt.edu.cn

Abstract. For the scenario of limited hardware resources and restricted software environment in edge computing architecture, the method of building and applying knowledge graph in edge analytics environment is proposed. The main process includes: building knowledge graph in the cloud, storing knowledge base with RDF format at the edge through customization and transformation, and performing query and analytics at the edge with SPARQL graph search language. The method is simulated in a communication anti-jamming test environment, and the results show that the relevant technical solutions can better meet the requirements of construction of knowledge graph and data analysis in a restricted edge analytics environment.

1. Introduction
Edge computing [1] is an emerging form of computing that complements cloud computing approaches by providing local environment awareness and response services, which can relieve the pressure of processing massive amounts of data in the cloud, eliminate the round-trip overhead to the cloud, and overcome the impact of cloud disruptions[2]. Combining edge computing with knowledge graph to provide edge analytics services is a new hotspot for applications such as IoT, with its ability to complete the data processing process in the vicinity of the computing location, thus providing fast data analytics support services. However, infrastructure resources such as CPU usage, memory capacity, network bandwidth and operating system of relevant devices in this scenario is usually restricted, and commercial graph database products such as Neo4j and nebula graph are often difficult to be deployed directly for applications. In this study, a practical knowledge graph construction and application method is proposed to address these problems, and the relevant method is simulated and verified in a communication anti-jamming test environment. With customized and transformed knowledge graph at the edge, the system can realize direct and fast decision making for known jamming patterns. For unknown jamming patterns the system can make inference decision based on the characteristic parameters of jamming patterns. The results show that our method can provide fast analytic data results based on the knowledge graph at the edge under restricted edge computing conditions.

2. Related work
Knowledge graph is an important technique for data integration and inference analysis, which uses graph models to describe knowledge and model the association relationships between things in the world, and is important for explainable artificial intelligence [3]. Knowledge graph consists of nodes and edges, which can be entities or abstract concepts, and edges can be entity attributes or
relationships between entities. The early idea of knowledge graphs originated from Semantic Web and is the result of the influence and development of related technologies such as ontology, knowledge representation, knowledge reasoning and natural language processing. Knowledge graph is currently an important supporting technology for data management and inference analysis, and has been widely applied and produced significant utility in cloud computing scenarios. With the continuous development of edge computing architectures, the application of knowledge graph in edge computing environments has gradually gained the attention of researchers. Typical research advances include:

In study [4], Anand N et al. proposed an approach for industry deployment where knowledge graphs are used to capture domain knowledge for reusability, maintainability, and data stream communication normalization for infrastructure instantiation. The approach decoupled the model knowledge from the development knowledge in a deployment and is presented through an in-flight use case from a large oil and gas client.

In study [5], Xu W et al. proposed SuccinctEdge, a compact, decompression-free, self-index, in-memory RDF store that can answer SPARQL queries, including those requiring reasoning services associated to some ontology. They provided details on its design and implementation before demonstrating its efficiency on real-world and synthetic datasets.

In study [6], Zhu P et al. proposed a defect diagnosis technology solution for power communication networks based on edge computing architecture combined with knowledge graph technology. The study incorporates multiple intelligent methods for defect diagnosis under edge computing architecture, and achieves timely defect detection and dispatch maintenance by analyzing device alarm logs and tracking network topology.

3. Architecture

With reference to the above research progress, a knowledge graph data analytics method based on edge computing architecture is constructed in this study to adapt for restricted infrastructure resources such as CPU usage, memory capacity, network bandwidth and operating system. The process of building and applying the knowledge graph is completed in the cloud and the edge respectively, and the architecture of the system is shown in Fig. 1.

**Fig. 1** Architecture of Building and Applying knowledge graph based on edge analytics

The architecture mainly includes two processes in the cloud and the edge. 1) In the cloud, various feature parameters are extracted from historical data sets through machine learning and subgraphs are generated, followed by knowledge fusion and other methods, combined with manual customization by
subject matter experts to generate domain-specific knowledge graphs and related analytic rules. Finally, the domain knowledge base is constructed in the cloud. 2) In the edge, the domain knowledge base is customized and converted to construct the knowledge base at the edge. When new data signals are generated from sensors at the edge, the relevant features of data signals are constructed as subgraphs to be matched, searched in the knowledge base at the edge according to the graph matching algorithm, and different analytics computation strategies are selected according to the matching results of the subgraphs.

4. Experiments and Results
In this study, experiments related to the construction and application of knowledge graph in edge analytics environments are carried out in a communication anti-jamming test scenario. During the experiments, this study uses a bottom-up approach in the cloud from the existing conventional jamming style data to extract entities and relationships from open linked data by machine learning or other technical methods, and extracts resource patterns from the openly collected data, selects new patterns with higher confidence among them, adds them to the knowledge graph after manual review, and then constructs top-level ontologies and decision rules to form an anti-jamming domain knowledge graph. The knowledge graph data in the cloud is relatively rich in content, and commercial graph database products (such as Neo4j) are used for storage.

At the edge, restricted by factors such as hardware resources and software environment, the anti-jamming knowledge base in the cloud is customized, transformed to RDF format, and is stored as knowledge base at the edge. RDF is a schema-free data model recommended by W3C, which takes the form of a graph consisting of a set of triples. Each triple consists of a subject, a predicate and an object. The attributes can be qualified as objects or data types. They all relate a URI (or a blank node) to a URI or a literal content, respectively.

For searching in the knowledge base at the edge, this study uses SPARQL syntax, another W3C recommendation, whose syntax is inspired by SQL’s SELECT-FROM-WHERE, using an approach based on matching Basic Graph Patterns (BGP), a set of triple patterns (TP) on an RDF graph, to retrieve the set of query answers. Figure 2 shows an example of the results obtained by searching the knowledge graph using SPARQL statements at the edge.

Fig. 2 Example of search results in knowledge base at the edge
The specific process of knowledge base search and data analysis at the edge is shown in Figure 3.
Fig. 3 Example of analytic data generation process at the edge

When searching the knowledge graph at the edge, by inputting data such as jamming pattern parameters, a sparql query statement is constructed which represents the anti-jamming strategy corresponding to the jamming pattern. Then the sparql statement is executed on the knowledge graph under the distinction between known and unknown jamming patterns, and then the decision parameters corresponding to the anti-jamming strategy are calculated by the algorithms realized in the system, and the anti-jamming decision results is finally generated. For known jamming styles, the anti-jamming strategy is obtained directly through the knowledge graph, and then the parameters are calculated according to the strategy; for unknown jamming styles, the jamming style features are calculated through the function, and the anti-jamming strategy is obtained in the knowledge graph according to the feature, and then the strategy parameters are calculated.

For example, inputting jamming pattern parameters of wide band block jamming, the corresponding anti-jamming strategy can be searched in the anti-jamming knowledge graph. In the case of perceptual error, the decision accuracy is shown in the Figure 4.

Fig. 4 Decision accuracy curves under perception errors
The ordinate is decision accuracy, which indicates whether the decision result based on knowledge
graph is correct. The abscissa is perception error rate, which represents the offset error of the perceived jamming parameters in frequency. As seen from the results, with the continuous improvement of perception error, decision accuracy decreases, while the decline is relatively gentle. The decision has certain robustness.

The experimental results show that the anti-jamming decision can be achieved by constructing the knowledge graph at the edge. For known jamming patterns, the system can realize direct and fast decision making; for unknown jamming patterns the system can make inference decision based on the characteristic parameters of jamming patterns. The decision parameters are also calculated. The system has good scalability in the experimental environment and smart decision-making ability for different jamming styles.

5. Conclusion

This study proposes a method which builds and applies knowledge graph in edge analytics architecture. The method mainly includes: building the knowledge graph in the cloud, storing the knowledge graph in RDF format at the edge after customization and transformation, and combining with SPARQL statements to query and assist analysis of data at the edge. This method is simulated and verified in the communication anti-jamming experimental scenario, and it can search and provide fast analytic data results based on the knowledge graph under the restricted edge computing conditions. In the subsequent work, the compressed storage of knowledge graph at the edge will be investigated in depth, and the current data analysis service will be further optimized in conjunction with the research progress such as knowledge inference.

Acknowledgments

This work is supported by the Jiangsu Province “333 Projects” (grant No. BRA2020418) and the Universities Natural Science Research Project of Jiangsu Province (grant No. 20KJB413003).

Reference

[1] Shi W and Dustdar S 2016 The Promise of Edge Computing Computer 49 p 78
[2] Ai Y, Peng M and Zhang K 2018 Edge computing technologies for internet of things: a primer Digital Communications and Networks 4(2) p 77
[3] Wang H, Qi G and Chen H 2019 Knowledge Graph (Beijing: Publishing House of Electronic Industry)
[4] Anand N, Chintalapally A, Puri C and Tung T 2017 Practical Edge Analytics: Architectural Approach and Use Cases IEEE International Conference on Edge Computing (EDGE) (Honolulu) (IEEE)
[5] Xu W, Curré O and Calvez P 2020 Knowledge graph management on the edge
[6] Zhu P, Cai X, Xu S, et al. 2021 Research of fault diagnosis of power communication network in edge cloud computing Application of Electronic Technique 47 p 30
[7] Han C, Niu Y T and Xia T 2020 Detection algorithm and anti-jamming method for linear sweeping jamming with low complexity Application Research of Computers 37 p 267 (in Chinese)
[8] Xia Z, Chen J and Niu Y 2019 Performance Analysis of a Detection Method for Follower Jamming in Wireless Communication Journal of Terahertz Science and Electronic Information Technology 17 p 5 (in Chinese)