Design and Implementation of Traffic Big Data Super Correlation System

Yan Yupeng[1,2], Ke Jiang[1,*],Cheng Jianpeng[2]

[1]Beijing Institute of Technology, Beijing, China
[2]China Changfeng Science Technology Industry Group Corp, Beijing, China

[*]e-mai: alston.hi@163.com

Abstract. In order to make full use of traffic big data resources and meet the comprehensive management needs of traffic management departments for traffic congestion, accidents and violations, a traffic big data super correlation system based on Knowledge Graph is designed and implemented. The system is based on business data such as people, vehicles, roads, enterprises, accidents, and violations, and uses Neo4J, MPP, and ES as data storage and search engines. By building 19 relationship models of 6 types of entities, the access storage and retrieval analysis of big data are realized. After deployment and testing, the system meets business requirements in terms of functions; in terms of performance, the overall data volume reaches a scale of 100 million, and the search response time meets performance requirements. It can provide traffic management agencies with a variety of practical application services with accurate results and comprehensive analysis.

Keywords: Knowledge Graph; Big Data; Traffic Management; Neo4J.

1 Introduction

With the development of big data technology, intelligent transportation mode has gradually become a new research direction [1]. In the application of intelligence research and judgment business, traffic management departments in various places already have information systems such as intelligent search and holographic archives [2-3], but traditional knowledge retrieval tools use full-text databases, relational databases, etc., the relationship query efficiency is low, the query method is not friendly, and it is difficult to meet the actual combat needs of traffic departments in traffic congestion management, clue verification, duty control, special project operations, etc. There is an urgent need for a big data super correlation tool to effectively correlate personnel's related vehicles, violations and accidents. At present, in the application of Knowledge Graph technology, it has been established by establishing reasonable associations between criminal actors to identify potential suspects, and using spatial, temporal, and modus operandi-based similarity matching to establish hierarchical associations between criminal entities [4-5].

M. Yu et al. (eds.), *Proceedings of the 2024 5th International Conference on Big Data and Informatization Education (ICBDIE 2024)*, Advances in Intelligent Systems Research 182,
https://doi.org/10.2991/978-94-6463-417-4_27

The traffic big data super correlation system uses graph database storage to build Knowledge Graph [6], which solves the performance bottleneck problem of traditional relational databases when storing relational data. Traditional relational databases use entities and relationships as the metadata of Knowledge Graph. When performing knowledge retrieval, complex association relationship queries are required, resulting in low query efficiency. Graph databases are different from traditional relational databases in that they are highly scalable and can provide rich relationship presentation methods. They can intuitively display Knowledge Graph, which supports adding, deleting and modifying graph structures. Compared with traditional relational databases, the storage method of graph databases is more suitable for efficient query of multiple associations, and can also realize more complex graph matching queries and reasoning through classical graph algorithms.

At the same time, the use of Knowledge Graph canvas technology can build vivid and intuitive relational views, simulate the way of thinking of the human brain for deduction and judgment [7], and dynamically deduce people, items, and events related to people in time and space dimensions based on a variety of algorithm models, which is convenient for managers to quickly mine case clues, find relevant targets, and implement control measures.

2 System design

2.1 System architecture design

The technical architecture of the transportation big data super-correlation system adopts the front and rear separation design. The back-end of the super-correlation application uses SpringCloud microservice to register various Knowledge Graph services, and the front-end uses Vue.js to show the relational performance and analysis of Knowledge Graph. In the storage layer, the graph database is stored using Neo4J, the big data storage uses MPP, the portrait data uses FTP, and the index data uses ES index database to realize the access storage and retrieval analysis of multi-source heterogeneous transportation big data.

2.2 System technology roadmap

The system completes knowledge acquisition by building a Knowledge Graph through a large-scale multi-source data storage graph data warehouse, which can realize the efficient, high-quality and rapid construction of the ternary relationship group of entity, relationship and attribute information [8]. At the same time, the update method and frequency of the knowledge base are different according to different types of entities and relationships, including the full and incremental update of data.

The portrait features of the target person or group are constructed through graph reasoning and graph mining methods, and the validity of the portrait is tested and verified by the feature comparison in the actual scene [9]. In the traffic scene, the user attribute information and travel information are used to construct the early warning integral model of traffic violations, and the early warning of suspected violations is verified by the test data fact comparison [10].

Construct a Knowledge Graph for human-machine collaboration. In terms of front-end development, it realizes smooth and smooth front-end controls such as dragging entities and relationships. At the same time, the graph deduction process ensures that entity points do not drift [11]. At the same time, the time dimension is added to the graph display, which can filter entities and relationships according to the time dimension [12]; intimacy analysis is added, and each type of relationship can be customized weight [13]; using the characteristics of the graph database, quickly find the shortest path and hidden relationships [14].

1.Knowledge Graph Construction

Knowledge Graph construction includes three parts: information extraction, knowledge integration, and knowledge update: 1) Information extraction: Extract entities, attributes, and inter-entity relationships from various types of investigation and case-handling clue data sources, and form ontological knowledge expression on this basis; 2) Knowledge integration: Integrate entities and relationships from multiple sources. Since most of the data required by this application comes from the relations of production data warehouse, it is mainly reflected in knowledge merger. The integration of the data layer, including the reference, attributes, relationships, and categories of entities, avoids the conflict of instances and relationships, resulting in unnecessary redundancy; 3) Knowledge update: Update the knowledge base by incremental coverage.

2. Knowledge Graph Storage

Knowledge Graph storage uses Neo4J [15]. Relational databases are only suitable for queries with relationships within two layers. When longer-range and wider-range relational queries are required, the function of a graph database is required. A graph database refers to a database that stores and queries data in the form of a graph data structure. Neo4J belongs to a native graph database. The storage backend it uses is specially customized and optimized for the storage and management of graph-structured data. The physical addresses of nodes associated with each other on the graph also point to each other in the database, so it can better play the advantages of graph-structured data. In Knowledge Graph, the organization of knowledge adopts a graph structure, so it is very suitable for storage with Neo4J. The advantages of graph databases are: 1) performance, fast query speed for long-range relationships; 2) good at discovering hidden relationships, such as by judging whether there is a path between two points on the graph, you can find the relationship between things.

The Knowledge Graph stored procedure is implemented by ETL tools such as Kettle [16]: 1) Use ETL tools to store the constructed Knowledge Graph data into text files and push them to the graph database server; 2) Write file parsing scripts to parse the files and store them in the graph database; 3) Compile API interfaces.

3. Knowledge Graph Display

The front-end display uses Vue.js [17-18] to display the graph. Based on the data lake visualization library, it supports a large number of user interaction scenarios with frequent applications, such as highlighting edges/nodes, expanding/retracting entities, changing the shape of nodes, adding nodes, deleting nodes/relationships, etc. 1) Initialize the Vue.js component; 2) Read the entity data and first-level relationship data through the interface; 3) Separate the node and relationship data according to the returned data; 4) Draw the node data and relationship data on the graph. This step mainly includes three types of operations: DOM operation, data binding and data processing.

3 System Implementation

The traffic big data super correlation system provides relationship calculation and automatic association between different object entities, organizes scattered data, establishes associations between data, reveals the inherent meaning of information, and discovers clues and actionable intelligence from a large amount of information. As shown in Figure 1, the system is composed of modules such as entity expansion, conditional expansion, event expansion, entity operation, relational operation, and graph operation

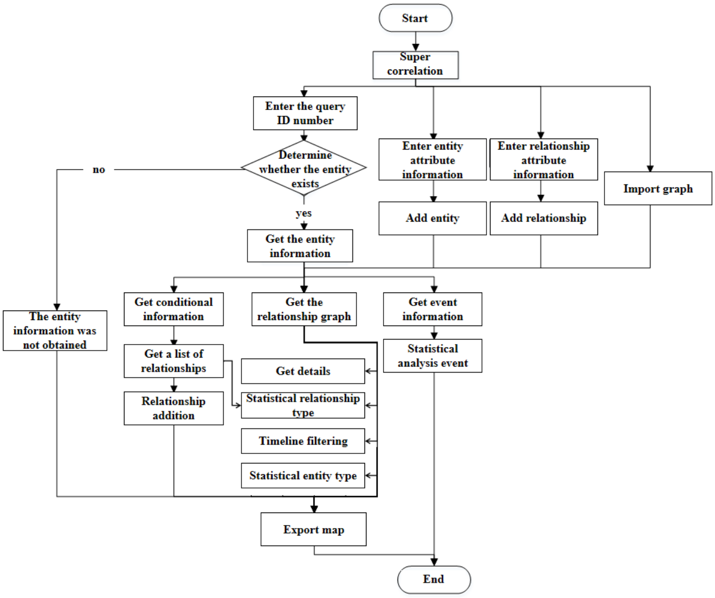

Fig. 1. Logical processing structure of traffic big data super correlation system

3.1 Entity Extension sub-module

Entity extension subsystem is an important part of super correlation, which is a prerequisite for the realization of functions such as statistical analysis, time axis filtering, and entity and relationship details. Entity extension mainly queries the relationship information between entities in the graph database, and performs statistical analysis, time axis filtering, and detail query on the query information. It involves 6 entities and 19 relationships, including level 1 relationship and level 2 relationship.

6 kinds of entities: People, vehicles, roads, enterprises, accidents, violations. 19 kinds of relationships: The first-level relationship includes people and cars, people and violations, cars and violations, cars and accidents, people and enterprises _ service,

people and enterprises _, people and places, cars and places; the second-level relationship includes the same travel, the same company, the same car violations, first entry, frequent passing, accompanying travel, serial and parallel travel, frequent congestion, frequent accidents, abnormal congestion points, frequent night travel.

The main functions of the entity extension module include 1) entity expansion: entity expansion includes the query function of the 1-level relationship and the 2-level relationship of the entity. The platform adopts the form of front-end and back-end separation. The back-end program not only provides service interfaces for the front-end, but also supports interface calls of external systems; 2) Statistical analysis: Statistical analysis includes entity type statistical analysis and relationship type statistical analysis, which is used to count all entity types and relationship types in the current page; 3) Timeline filtering: Timeline filtering refers to the relationship graph in the current page. According to the entity time and relationship time, the entity and relationship information that meet the time range are highlighted with the movement of the time axis; 4) Details retrieval: Details retrieval includes entity details query and relationship details query. Relationship details query the information and relationship type statistics of the specified relationship type between two entities, and display the relationship details in chronological order; Entity details query entity information and entity-related relationship statistics. These two kinds of detail interfaces are in the form of front-end and back-end separation. The back-end program not only provides service interfaces for the front-end, but also supports interface calls from external systems.

3.2 Conditional Extension sub-module

The conditional extension subsystem is a supplement to the functions of the entity extension subsystem. The conditional extension queries the small-scale relationship graph of the entity according to a variety of conditions such as relationship type, relationship number and relationship time range. The main functions of the conditional extension include 1) Conditional extension: Includes the query function of the 1-level relationship and the 2-level relationship of the entity. The main query is the entity relationship graph that satisfies the conditions. The platform adopts the form of front-end and back-end separation. The back-end program not only provides the service interface for the front-end, but also supports the interface call of the external system. 2) Relationship details: The relationship details query the information of the specified relationship type between the two entities. The interface adopts the form of front-end and back-end separation. The back-end program not only provides the service interface for the front-end, but also supports the interface call of the external system; 3) Relationship addition: Selecting one or more relationships from the relationship list of the conditional extension to add to the relationship graph of the entity; 4) Timeline filtering: Highlighting the entity and relationship information that meets the time range according to the movement of the entity time and relationship time on the current page.

3.3 Event Extension sub-module

The event expansion subsystem is a query for entity event information, involving event information related to people, vehicles, roads, enterprises, accidents, and violations. The event expansion subsystem includes event expansion, event type statistics and other modules. The main functions of event expansion include 1) event expansion: The main function module of event expansion is to query the Gbase database to obtain the entity event information list according to the entity label, entity attribute and entity attribute value; 2) event type statistics: The event type statistics perform statistical analysis on the event type of the subject.

3.4 Entity Operation sub-module

Add people, cars, roads, enterprises, accidents, and violations to the graph, and do uniqueness verification on them, and delete entities in the graph. Users enter a series of attributes of a certain entity, do uniqueness verification on its unique field, and then add the entity to the map of the canvas, and the specified entity can also be deleted in the map.

3.5 Relational Operations sub-module

Add 8 level 1 relationships such as people and cars, people and violations, cars and violations, cars and accidents, people and enterprises _ service, people and enterprises _ ownership, people and places, cars and places to the map; 11 level 2 relationships such as same travel, same enterprise, same car violations, first entry, frequent passing, accompanying travel, serial and parallel travel, frequent congestion, frequent accidents, abnormal congestion points, and frequent night travel. The user selects two entities in the canvas, selects the relationship to be added, enters the attribute information of the relationship, and then adds the relationship to the map of the canvas.

3.6 Graph Operation sub-module

Users can save maps to local or personal libraries, view the relevant information of personal libraries and shared library maps, import maps into the canvas, share their own maps with others, and collect shared maps. If the user wants to save the current map, click the Save Map button, you can choose to save to personal space or save to local, if you choose to save to personal space, enter the relevant information of the map, click the Save button, and enter the personal research and collaborative sharing page. Users can query the maps of personal research and collaborative sharing according to conditions. Users can share the maps of personal libraries with other users, or collect the maps of collaborative sharing. Users can import maps from personal libraries and shared libraries into the current canvas respectively.

4 System test

The system can be applied to multiple industries, and a specific type of big data scenario is selected for system testing. According to "GB/T 25000.51-2016 System and Software Engineering", System and Software Quality Requirements and Evaluation (SQuaRE) Part 51: Quality requirements and test rules for ready-to-use software products (RUSP), a test system is built, and the industry simulation data is used to verify, and 23 kinds of relationships between people, mobile phones, vehicles, enterprises, addresses, and cases are tested. A total of 9 functional use cases and 1 performance use case are set.

4.1 Testing environment preparation

Each module of the system adopts the cluster deployment mode, the front and back end perform message communication based on TCP-IP protocol, the data transmission of the back end is based on Kafka message queue, the original service data is indexed into ES for fuzzy search, the Knowledge Graph analysis results are stored in MySQL, and the traffic management service data is pre-processed and cached into Redis in preparation for the relationship mining algorithm to quickly read and call. The system deployment operation is shown in Figure 2; the system deployment configuration is shown in Table 1.

Table 1. System deployment configuration

Serial number	Purpose	Device configuration	Deployment software
1	Application Services Virtual Machine * 3	Virtual Machine: CPU Intel(R) Xeon(R) E5-2658A v3 @ 2.20GHz, 8-core memory: 8GB system disk: SAS 40GB data disk: SAS 800GB	Operating System: CentOS 7.0 Application Container Engine: Docker 1.11.2 middleware: Nginx 1.11.8 database: PostgreSQL 12.9, MysqlServer 8.0.19, ElasticSearch6.5.1, Neo4j 9.6.2
2	Cache Middleware Virtual Machine*2	Virtual Machine: CPU Intel(R) Xeon(R) E5-2658A v3 @ 2.20GHz, 8-core memory: 16GB system disk: SAS 40GB data disk: SAS 140GB	Operating System: CentOS 7.0 cache component: Redis 3.0.7.9 database: PostgreSQL 12.9, MysqlServer 8.0.19, ElasticSearch6.5.1, Neo4j 9.6.2
3	Data Storage Middleware Virtual Machine*3	Virtual Machine: CPU Intel(R) Xeon(R) E5-2658A v3 @ 2.20GHz, 8-core memory: 8GB system disk: SAS 40GB data disk: SAS 2.3TB	Operating System: CentOS 7.0 message component: Kafka 2.11-1.1.0, Zookeeper 3.4.10 database: PostgreSQL 12.9, MysqlServer 8.0.19, ElasticSearch6.5.1, Neo4j 9.6.2
4	Microservices Aplication Virtual Machine *3	Virtual Machine: CPU Intel(R) Xeon(R) E5-2658A v3 @ 2.20GHz, 8-core memory: 8GB system disk: SAS 40GB data disk: SAS 140GB	Operating System: CentOS 7.0 Application Container Engine: Docker 1.11.2 Microservices Engine: Spring CloudFinchley.SR1/Spring boot 2.0.4 Power builder: IntelliJ IDEA

5	API Gateway Virtual Machine*6	Virtual Machine: CPU Intel(R) Xeon(R) E5-2658A v3 @ 2.20GHz, 8-core memory:16GB system disk：SAS 40GB data disk：SAS 140GB	Operating System: CentOS 7.0 message component: Kafaka 0.10.1 database: PostgreSQL 12.9, MysqlServer 8.0.27, Spring Cloudgateway gateway service
6	Test end point	Virtual Machine: CPU Intel Core i7-7700K @4.20GHz 4.20GHz, 8-core memory: 8GB Hard disk: 600GB	Operating System: Windows 7 Professional Edition 64-bit Browser: Google Chrome 75.0.3770.100

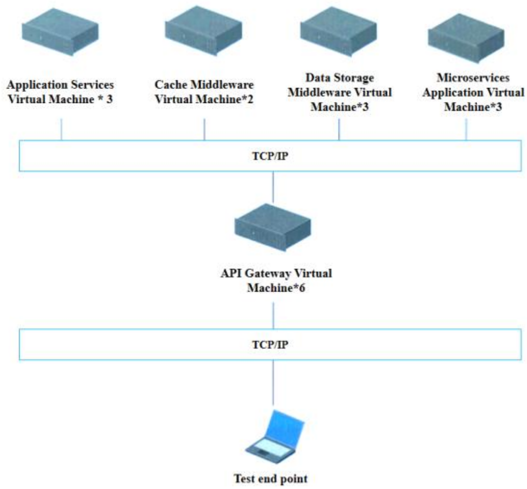

Fig. 2. traffic big data super correlation system test topology diagram

4.2 System function test

System function testing is mainly to perform UI automation testing on super-related web applications and save the return results of each server to generate txt files for business verification to detect whether the functions of the system are realized and the running effect. Table 2 is a list of 9 functional use cases of web applications.

Table 2. System Functional Use Cases

Serial number	Use case name	Use case description	Test result
1	Direct relationship	The inspection system enters the person's ID number and displays the function of the person's direct relationship graph of 6 types of entities (including people, mobile phones, vehicles, enterprises, addresses, and cases)	Passed
2	Dynamic deduction	Examine the dynamic deduction of the relationship between six types of entities in the entity direct relationship graph of the system, including 7 direct relationships (first-level relationships); functions with 16 second-level relationships	Passed

3	Event extension	Examine the system to expand the event of the entity, select the query start and end time, and display the function of the entity event list within the time range	Passed
4	Intimacy analysis	Examine the system to select personnel entities for intimacy analysis, and calculate the function of intimacy according to the number of occurrences of model relationships	Passed
5	Hidden relation retrieval	Examine the system's ability to select two entities and analyze hidden associations between them	Passed
6	Entity and relationship editing	Examine the functions of the system to add entities, add relationships, and reverse select, delete, and delete entities	Passed
7	Relational graph import and export	Examine the function of system to import/export entity relation graph in JSON format	Passed
8	Statistical analysis of relationship graph	Examine the function of the system for statistical analysis of relational graph according to entity type and time	Passed
9	Individual research and collaborative operations	Examine the system's ability to save or share relationship graphs with other accounts	Passed

4.3 System performance testing

The system performs performance testing in a big data environment. The system performance test is to monitor the response time of the system searching for entity information while conducting UI automation testing. Table 3 shows the average response time of the performance test.

The test results show that when the business data of this system reaches 100 million, the response time of the entity query function reaches 2.313 seconds, and the test data has exceeded 100 million, which can meet the business requirements.

Table 3. System performance use cases

Serial number	Use case name	Test requirements	Test Result
1	search response time	Under the condition of billion-level data scale, perform super retrieval, and the retrieval response time does not exceed 5 seconds	A total of 100,565,183 simulated data from the permanent population information database. The fuzzy search of names on the hyper search page: view the browser's return page response time, with an average of 2.313 seconds

5 Conclusion

The traffic big data super correlation system is based on the application of Knowledge Graph technology to display, deduce and analyze the relationship between entities in an intuitive and dynamic way. Knowledge Graph deeply mines the association relationship from big data, which can analyze hundreds of millions of massive relationship data in quasi-real time and convert it into relationship graph data. It supports traffic management users to carry out intelligence research and judgment analysis. It provides strong support for the association clue mining of personnel, vehicles and other targets. It assists in the relationship intimacy analysis, finding hidden relationships and other event deduction, just like the case of vehicles, serial and parallel case vehicle information query. This system has the advantages of high timeliness, strong practicability, accurate recommendation results, and comprehensive analysis. The summary is summarized into the following three aspects:

1. Analysis means: Different from the existing big data system mainly based on rule query means, the system has established a large number of business-related intimacy analysis, relational analysis model library, algorithm library to analyze and mine data value, mining hidden case clues.

2.Timeliness: Compared with the existing big data system is mostly after-the-fact query analysis, the system can perform real-time analysis, so as to achieve advance warning and timely control.

3. Practicality: Compared with simple data retrieval, the system combines traffic people, vehicles, roads, enterprises, accidents, violations of the subject data for relationship analysis, more comprehensive analysis.

The system can be applied to traffic congestion management and illegal control, making full use of the huge traffic data, providing more comprehensive, accurate and intelligent application services for traffic management congestion management, abnormal investigation, vehicle control and other work.

References

1. Sussman J M .AN INTRODUCTION TO INTELLIGENT TRANSPORTATION SYSTEMS[J]. 2005.
2. WANG Li-bo, LIU Yu-xuan. Research on Application of Intent Recognition and Semantic Slot Filling Joint Model in traffic Search System[A].China Computer Users Association Network Application Branch, China Computer Users Association Network Application Branch Paper of the 25th Annual Conference on New Network Technologies and Applications in 2021[C]. Department of Information and Network Management, People's traffic University of China; Department of Finance, People's traffic University of China; 2021:401-407.
3. Liang Changming, Gong Yan, Xue Hao et al. On the Intelligent Retrieval Technology of Public Security Massive Data Based on Distributed Search Engine [C]//"Smart City" Magazine, US-China Journal Academic Exchange Association. Proceedings of the 2016 International Academic Exchange Symposium on Smart City and Informatization Construction IV. Research and Development Department of Science and Technology Department of Shanghai Public Security Bureau, 2016:2.
4. Zhang Meijing. Application of Knowledge Graph in Criminal Intelligence Analysis [J]. 2021.
5. Qazi N, Wong B L W. Behavioural & Tempo-Spatial Knowledge Graph for Crime Matching through Graph Theory [C]//2017 European Intelligence and Security Informatics Conference (EISIC).IEEE Computer Society, 2017.

6. Liu Qifeng, Wang Jineng, Sun Min. Analysis of Knowledge Graph Construction Based on Big Data Technology [J]. China Security, 2023, (09): 31-35.
7. LU Feng, LI Lingling. Research on the Construction and Application of Criminal Suspects' Social Relations Based on Knowledge Graph[J].Journal of People's Public Security University of China (Natural Science Edition), 2023,29(02):94-100.
8. Shen Yunfeng. Construction and Intelligent Application of Public Security Knowledge Graph Model Based on Multi-source Heterogeneous Data[J]. traffic research, 2021(5):11.
9. Nicola Lettieri; Alfonso Guarino; Delfina Malandrino & …Rocco Zaccagnino. Knowledge mining and social dangerousness assessment in criminal justice: metaheuristic integration of machine learning and graph-based inference[J]. Artificial Intelligence and Law, 2023, Vol.31(4): 1-50.
10. Park, Roy C; Hong, Ellen J. Urban traffic accident risk prediction for knowledge-based mobile multimedia service.[J].Personal & Ubiquitous Computing,2022,Vol.26(2): 417-427.
11. QIU Rui, ZHU Zhen-hua. Application of Knowledge Map in Public Safety[J].Computer Knowledge and Technology, 2018, 14 (35): 196-199.
12. ZHU Zhen-hua, YU Xiao-yun, LI Chao. Research on Prediction Method of Personnel Relationship Based on Knowledge Map[J].Computer Knowledge and Technology, 2018, 14 (28): 176-178.
13. Chen Zhonghong, Social gang relationship map analysis system based on intimacy quantitative model. Anhui Province, iFLYTEK Information Technology Co., Ltd., 2015-09-24.
14. Hou Jiaqi. A Study on Entity Relation Extraction Based on Deep Learning [D]. People's Public Security University of China, 2021.
15. MA Fujian, LI Xiwei, HUANG Wenli. Construction of Bogie Knowledge Base Based on Knowledge Map Neo4j[J]. Journal of Dalian Jiaotong University, 2023, 44 (05): 79-85.
16. WEI Ya-jun, ZHANG Wen-wen, LI Dong-qing. Research on the Data Conversion Synchronization Method Based on Kettle [J].Software Guide, 2022, 21(08):126-131.
17. KANG Shanshan, LIU Li, TIAN Ling, TAN Haotian, LUO Bingjun. Development of Laboratory Management System Based on Front-end Framework Vue. js[J]. RESEARCH AND EXPLORATION IN LABORATORY, 2023, 42 (03): 281-284+318.
18. Su Peicheng. Research on visualization of social network relations based on D3.js [J]. Electronic Technology and Software Engineering, 2021, (03): 205-206.