Analysis of Disease Data Based on Neo4j Graph Database

Jian Zhao, Zhiguo Hong*, Minyong Shi
School of Computer Science and Cybersecurity
Communication University of China
Beijing, 100024, China
hongzhiguo@cuc.edu.cn

***Abstract*—As we all know, there are many diseases in the natural environment in which we humans live. A disease may show a variety of symptoms in patients, such as appendicitis, a variety of symptoms of appendicitis including abdominal pain, fever, gastrointestinal reactions and so on. At the same time, a symptom may also correspond to a variety of diseases. The relational database uses a large number of links to represent and query these complex and larger correspondences, which is very expensive. Many data relationships in the real world are graphical, and the graph database can better describe such data [1].The Neo4j graph database's data store has pointers to their neighbors, so it's easy to extend the newly discovered content. In addition, querying highly correlated data is very efficient for the Neo4j graph database. In the graphical database, the symptoms associated with each disease and the association between the disease and the disease can be clearly displayed to help people better judge the disease. This paper first introduces the data into the Neo4j graph database, and then introduces Neo4j's query language Cypher, so as to make a more intuitive image analysis and provide corresponding treatment suggestions for the disease data of related queries.**

Keywords—Disease data; Neo4j; Relational Database; Cypher

I. INTRODUCTION

With the advancement of society, people's living standards are getting higher and higher, and the conditions for food, clothing, housing and transportation have been greatly improved. These changes have indirectly led to an increase in the number of diseases. In the early 19th century, like hypertension, hemiplegia was rare, and now we have to face these diseases. There are also inextricable relationships between many diseases, such as the symptoms we mentioned earlier. In addition, we will give treatment recommendations for each disease (such as recommending relevant hospitals or recommending related drugs after diagnosis). If we store these data in a relational database, we use foreign keys to implement links between different tables. When the amount of data increases and the relationship between data and data becomes more and more complicated, the traditional relational database not only generates a large amount of data redundancy, but also dynamically updates. In the graph database, we can clearly see the causal relationship between disease and disease, and can prevent it in time to prevent further deterioration of the disease. For example, an increase in rhinitis can lead to the occurrence of pharyngitis.

II. GRAPH DATABASE-NEO4J

In the development of database technology, there have been many data models. There are three kinds of commonly used in these models, which are hierarchical model, graph model and relational model. The relationship model is supported by a strict mathematical foundation, with high data independence and security, and simple operation.Currently relational databases are the most widely used data storage technology. However, with the continuous innovation of Internet technology, the scale of data on the network continues to increase, and the complexity of data is increasing. The relationship model has been slowly unable to meet the needs of related fields, and more and more problems appear in relational databases. The emergence of the graph database effectively alleviated this phenomenon.

The graph database is a new NoSQL database based on graph theory [2].Its data storage structure and data query methods are based on graph theory. In the graph calculation, the basic data structure expression is: G = (V, E), V = vertex, E = edge. In the graph database, the data model is mainly represented by nodes and edges [3], and the key-value pairs can also be processed. Has the following characteristics:

- Contains nodes and edges
- Attributes (key-value pairs) on the node
- The side has a name and direction, and always has a start node and an end node
- Edge can have attributes

The Neo4j database is a high-performance Nosql graph database written by Java and Scala, dedicated to the storage of network graphs. As a graph database, Neo4j has the following advantages:

- Faster database operations
- More intuitive data
- More flexible
- The speed of database operations does not decrease significantly as the database grows.

* Corresponding author
Email address: hongzhiguo@cuc.edu.cn(Zhiguo Hong)

978-1-7281-0801-8/19/$31.00 ©2019 IEEE
ICIS 2019, June 17-19, 2019, China

- Self-contained query language (called Cypher)
- The structure of the entity relationship is very natural and fits the intuitive feeling of human beings.

III. Create Neo4j Disease Database

As we introduced in II , the Neo4j disease database [4] consists of nodes and relationship attributes. The nodes are similar to object instances, and various relationships can exist between different nodes. Creating a Neo4j database is not difficult. The following is a detailed description of the Neo4j database for establishing appendicitis and its symptomatic treatment options.

In this example, a total of three types of nodes are set: disease node, symptom node, and treatment node. In Neo4j, tags can be used to identify a set of nodes. Through the operation of the set of nodes, we can achieve the purpose of indexing, defining constraints, and querying. "CREATE(n:Disease{name:'Appendicitis',type:'surgery'})return n;",Create a disease called Appendicitis, Data Surgery, and use the above statement to create a set of symptoms and treatment labels. " MATCH (n: Disease{name:'Appendicitis'}),(m:Dasease{name:'AcuteAppendicitis'}) CREATE(n)- [r:branch]->(m)" This indicates the presence of appendicitis Acute appendicitis and chronic appendicitis [5].

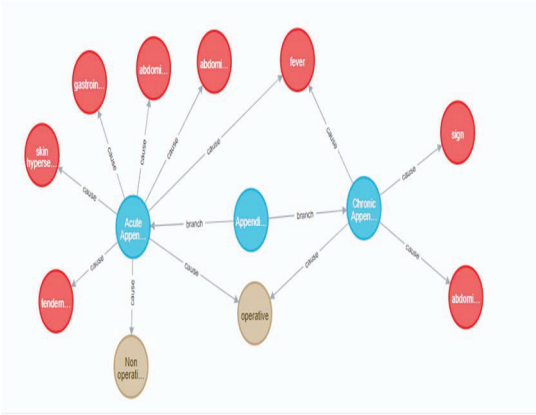

Fig. 1. Symptom division of appendicitis

IV. Disease Data Query and Analysis

When a disease database is created, it is possible to start querying and analysising the disease data present in the database [6]:First, each node represents a disease, and analysis of a single node can determine whether the disease is a property value of a disease such as surgery or internal medicine; Second, a disease has multiple symptoms and multiple treatments. This causal relationship can be clearly expressed by analyzing the relationship between two or more nodes; Thirdly, it can be analyzed to find the connection between different diseases, for example, with the same symptoms, to help people better distinguish the disease, and to get reasonable treatment as soon as possible.

A. CYPHER

"Cypher" is a descriptive graphical query language that allows expressive and efficient queries for graphics storage without having to write graphical structure traversal code. There are several obvious parts in this query language:

- START: At the beginning of the graph, it is obtained by the ID or index of the element.
- MATCH: The matching mode of the graphic, bound to the starting point.
- WHERE: Filter conditions.
- RETURN: Returns what is needed.

Look at the three keywords in the following example:

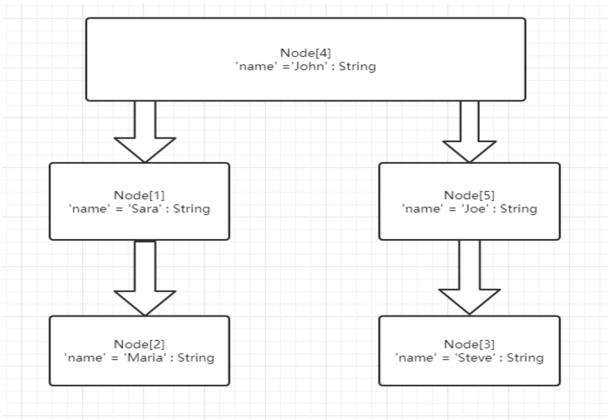

Fig. 2. Cypher example

There is a query that finds a friend of a friend named John (not his direct friend) in the index by traversing the graph, returning John and the friend of the found friend.

START john = node_auto_index(name = 'john')
MATCH john-[:friend]->()-[:friend]->fof
RETURN john fof

The results are returned as in Table I :

TABLE I. Return result 1

john	fof
Node[4]{name->"John"}	Node[2]{name->"Maria"}
Node[4]{name->"John"}	Node[3]{name->"Steve"}
2 rows,73ms	

Then add the filter: In the next example, list the IDs of a group of users and traverse the graph to find these users to take out the friend relationship line, returning the user with the attribute name and whose value starts with S.

START user = node(5,4,1,2,3)
MATCH user-[:friend]->follower

WHERE follower.name = ~/s.*/
RETURN user,follower.name
The results are shown in Table II :

TABLE II. Return result 2

user	follower.name
Node[4]{name->"John"}	"Steve"
Node[4]{name->"John"}	"Sara"
2 rows,4ms	

In this paper, five diseases that cause fever are used as an example. A relationship consisting of 26 nodes and 28 relationships is created. As shown in Fig.3, red represents disease, blue represents symptoms, and green represents treatment.

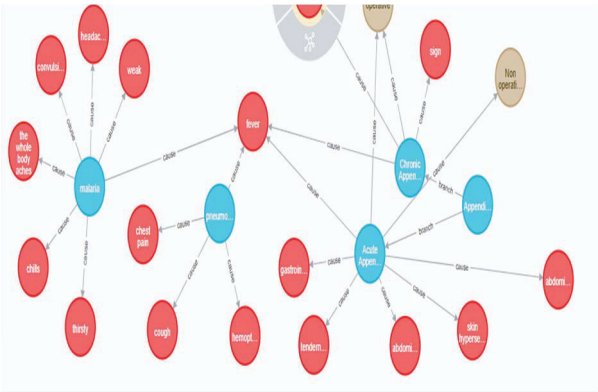

Fig. 3. Disease data map

B. *Query A Single Disease Node To View Its Attribute values*

Sometimes when patients go to the hospital to register, they often don't know which department they should go to. They use the Cypher statement to search for the disease you want to view. You can clearly see the department you should go to in the properties:

MATCH(n:Dasease{name:'malaria'}return n;Results are shown in Fig.4.

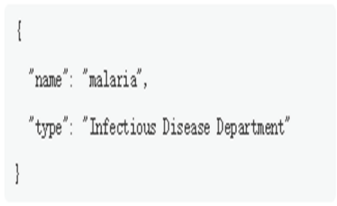

Fig.4 Single disease node query result

C. *Qurey The Symptoms And Recommended Treatment Of A Single Disease*

A disease has many symptoms. With the continuous development of medicine, more and more therapies have been continuously developed. The one-to-many relationship can be clearly expressed through the graph database.
MATCH(n:Dasease{name:'Acute Appendicitis'})-[:cuase]->(Symptom) return n,Symptom; Result is shown in Fig 5.

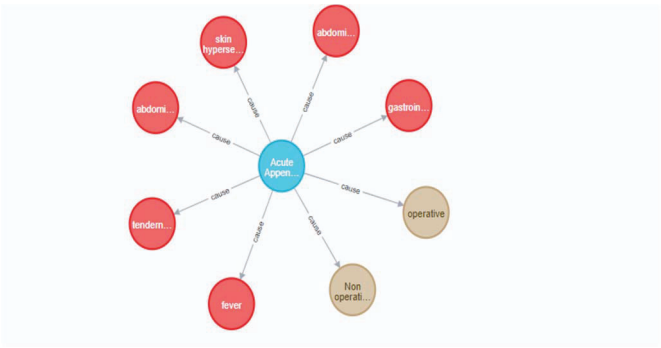

Fig.5 Single node disease symptom treatment search results

D. *Query Related Diseases According To Symptoms*

A variety of diseases can produce the same symptoms. You can enter a clear symptom to screen out the possible diseases and eliminate them according to other symptoms to achieve the auxiliary diagnosis.
MATCH(n:Symptom{name:'fever'})-[*1..2]-(Disease)return n,Disease; result is shown in Fig 6.

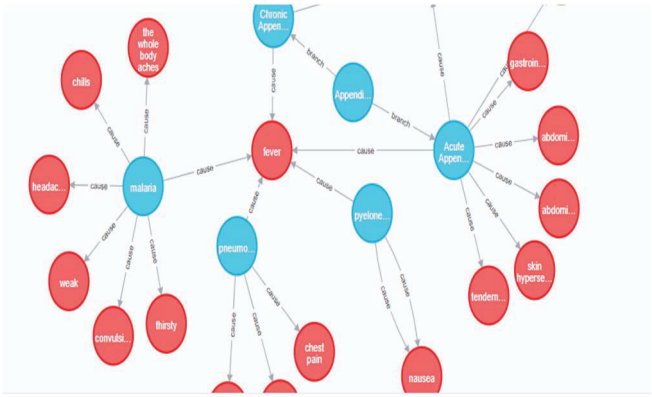

Fig.6 Search for relevant disease results based on symptoms

V. Conclusion

The Neo4j database is especially suitable for large, complex, low-structured data [7].Symptoms of the disease can vary from treatment to treatment, and in addition, suggestions for the rehabilitation of related diseases can be added to the data. Retrieving related diseases for the same symptom and clearly showing their correspondence better helps people overcome the disease

REFERENCES

[1] Lan Yu. Comparative Study of Graphic Database Neo4J and Relational Database[J]. Modern Electronic Technology,2012,35(20):77-79.

[2] Ye Tong. System design and development base on NoSQL database[D]. Nanjing University of Posts and Telecommunications,2018.

[3] Yaodong Cheng, Jianchang Zhao, Jun Xu. Research on Application Technology of Graphic Database[J]. Engineering Graphics,2006,27(1):143-148.

[4] Fenglin Qu. Research on health medical knowledge push system base on knowledge map[D]. Hainan University, 2018.

[5] Yongcang Wen. Comparison of CT and ultrasound diagnosis of acute and chronic appendicitis[J]. World Medical Information Digest,2017,17(15):125-129.

[6] Xue Li. Research and Implementation of Fuzzy Query Based on Neo4j Graph Database[J]. Computer Technology and Development,2018,28 (11):16-21.

[7] Yuanbo Ao, Zhiyong Hu. A model for MySQL data migration to Neo4j database[J]. Inner Mongolia Science and Technology and Economy, 2018(03) :90-92.