International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I2P125
Volume 73 Issue 2, 271-283, February 2025

Original Article

Graph Database with Neo4j and the Cypher Language: An Application in Mining Companies

Santiago Domingo Moquillaza Henríquez[1,2], Edwin Roque Tito[3], Juan Faustino Infantes Loo[4,5], Liliana Ruth Huamán Rondón[4]

[1]*Facultad de Ingeniería, Universidad Tecnológica del Perú, Lima, Perú*
[2]*Departamento Académico de Ciencias de la Computación, Universidad Nacional Mayor de San Marcos, Lima, Perú*
[3]*Academic Department of Systems Engineering, Universidad Nacional de Cañete, Lima, Peru*
[4]*Facultad de Ciencias Contables, Universidad Nacional del Callao, Lima, Perú.*
[5]*Escuela Superior Idat, Lima, Perú.*

[1]*Corresponding author : smoquillazah*@unmsm.edu.pe

Received: 03 October 2024 Revised: 06 January 2025 Accepted: 13 January 2025 Published: 21 February 2025

Abstract - *Objective of the Article: demonstrate how Neo4j, a graph database, together with the Cypher language, can be used to model and analyze complex relationships in the mining industry. It seeks to illustrate how this technology allows a clearer and more efficient view of mining operations and their global interactions. Applied Methodology: The mining system is modeled using graphs, where nodes represent key entities such as mining companies, mines, mineral types, export countries and transportation routes. The edges between these nodes reflect relationships, such as the extraction of minerals in specific mines, exports to countries, and the type of mineral transported (ferrous or non-ferrous). Cypher queries are used in Neo4j to explore and visualize these relationships. Results: The Neo4j implementation offers detailed and dynamic views of mining operations, facilitating the identification of critical patterns and relationships. Graphical visualization highlights connections and dependencies that may not be evident in traditional data models. Graph theory is applied to analyze connectivity and centrality in the data network, improving understanding and strategic decision-making in the mining sector. In summary, the paper shows the applicability of Neo4j in the mining sector using the Cypher Language in the context of graph theory.*

Keywords - *Neo4j, Mining sector, Graph database, Minerals, Cypher language.*

1. Introduction

It is important that no information has been found specifically applied to mining companies using graph databases. This article will see its applications in graphical databases using NEO4J with Cypher applied to Mining in Peru. The following algorithms are seen to enter into context the graph database; these antecedents of graph theory apply to all the current applications. Euler cycle. - If n nodes are of even degree, there is a Euler cycle. Visit all the edges, starting from a node and, reaching it and going through all the edges.

A Euler loop can help better plan mining companies' transportation routes by ensuring that each path is used only once, saving money. Hamilton cycle. -Traverse all the nodes starting from an origin and arriving at it. Applied to tour all mineral distribution centers. Warshall's algorithm to know if there is a path between one node and another The Floy-Warshall Algorithm, which this algorithm attempts to solve, is to find the shortest path between all pairs of nodes or nodes in a graph. Dijkstra's algorithm. Find the minimum distance between a starting node and an ending node. Finding the best route from the headquarters or branch to another Mina superintendency is very useful. Map Coloring. - An edge is drawn between adjacent areas to assign a different color or label; very useful for assigning products that cannot be together or adjacent to areas on maps where you want to use the minimum number of colors.

For example, in this code, to differentiate the various adjacent mining companies, assigning them a different color will be seen later.

- Depth-First Search (DFS): This visits the nodes based on a queue structure to traverse the nodes.
- Breadth-First Search (BFS): This visits the nodes based on a stack structure to traverse graphs.
- Prim's Algorithm: This algorithm finds the minimum spanning tree in a weighted graph. Choosing the node, it goes through the closest adjacent nodes, but considering the recalculation, without considering the cycles, until the tree is formed.

- Kruskal Algorithm: This algorithm, like Prim's, finds the minimum spanning tree in a weighted graph, but it works differently than Prim, which chooses the edges with the lowest weight, considering there are no loops.

- Centrality Algorithms: That allows you to identify the most important nodes of a graph.[1]

- Community Algorithms: A key feature of graphs showing real systems is how they are grouped into clusters, called community structures. [2]. These methods are mainly used in network analysis to find groups of nodes closely connected to each other compared to other nodes in the network.

It consists of finding a collection of groups from a set of elements that maximizes the relationship between the elements that make up each group. [3]

For example, the following code filters to see mines with common characteristics; this is a list of common minerals and extraction methods, along with aggregate statistics (average tons per day, extraction cost, and number of mines) for each group of mines that share those characteristics.

MATCH (m:Mina)-[:PRODUCE]->(min:Mineral), (m)-[:USO_METODO]->(me:MetodoExtraccion) WITH m, min, me, m.toneladas_diarias, m.costo_extraccion ORDER BY m.toneladas_diarias, m.costo_extraccion WITH min, me, COLLECT(m) AS minas WHERE SIZE(minas) > 1 // RETURN min.descripcion_mineral AS mineral, me.metodo_extraccion AS metodo, AVG(m.toneladas_diarias) AS avg_toneladas, AVG(m.costo_extraccion) AS avg_costo, COUNT(minas) AS cantidad_minas ORDER BY cantidad_minas DESC LIMIT 10

A graphical summary of what is used in graph theory is shown in Figure 1.

Fig. 1 Summary of the algorithms used by Neo4j

2. Literature Review

2.1. What are Graph Databases?

Graph database models are those in which data is organized and manipulated using structures based on graph theory.[5] Graph databases stand out as an innovative category designed to efficiently store and model interconnected data.[6] A graph database is an online data manager with Create, Read, Update and Delete (CRUD) capabilities [7].

2.2. What is Neo4j?

Neo4j is a graph database management system that allows modeling, storing and querying data in graphs; this technology is useful for domains with complex and interconnected relationships [8].

2.3. What are the Advantages of Graphical Databases Over Relational Databases?

Relational databases are good for managing transactional data and are widely used. However, with data growth with many relationships (for example, social networks), graph databases are becoming a good option for a data manager. data model. [9] A graph is a data structure composed of edges and nodes. Nodes have properties and connections as relationships, which also have properties. A transversal navigates a graph and identifies paths that order nodes. Graph databases constitute an important area of academic research and are also applied in industry. They are used to store, query and analyze many data types in different areas of business and academia. [10] The authors Qiu et al. developed a computer scheme to obtain semantic data from geological profiles in geological reports. Using Neo4j, they developed a knowledge graph database that incorporated vectorized geological profile data and related contextual text.[11] Their study demonstrated how graphical databases can improve understanding of the relationships between geological objects and stratigraphic evolution. There is a gap in the literature regarding the specific use of Neo4j for the comprehensive analysis of the mining sector in Peru. Our study seeks to fill this gap, providing a novel approach to analysing the main Peruvian minerals and their relationships with other key players in the sector. A review of the summarized literature is made by answering the following questions:

2.4. What are the Advantages of Graphic Bases?

The graph database did better at handling structural type queries than the relational database.[12] In general, graph databases achieve better results when tested objectively. This means that graph databases can retrieve results for predefined sets of queries faster than relational databases.[13] Nowadays, graphs have gained great popularity in social network analysis, healthcare, natural sciences, Business Intelligence (BI), and computer networks. Graphical Databases (GDB) facilitate the query and retrieval of complex graphical structures in a simple and fast way, which is difficult to model in traditional information systems based on relational databases (DBMS).[14]

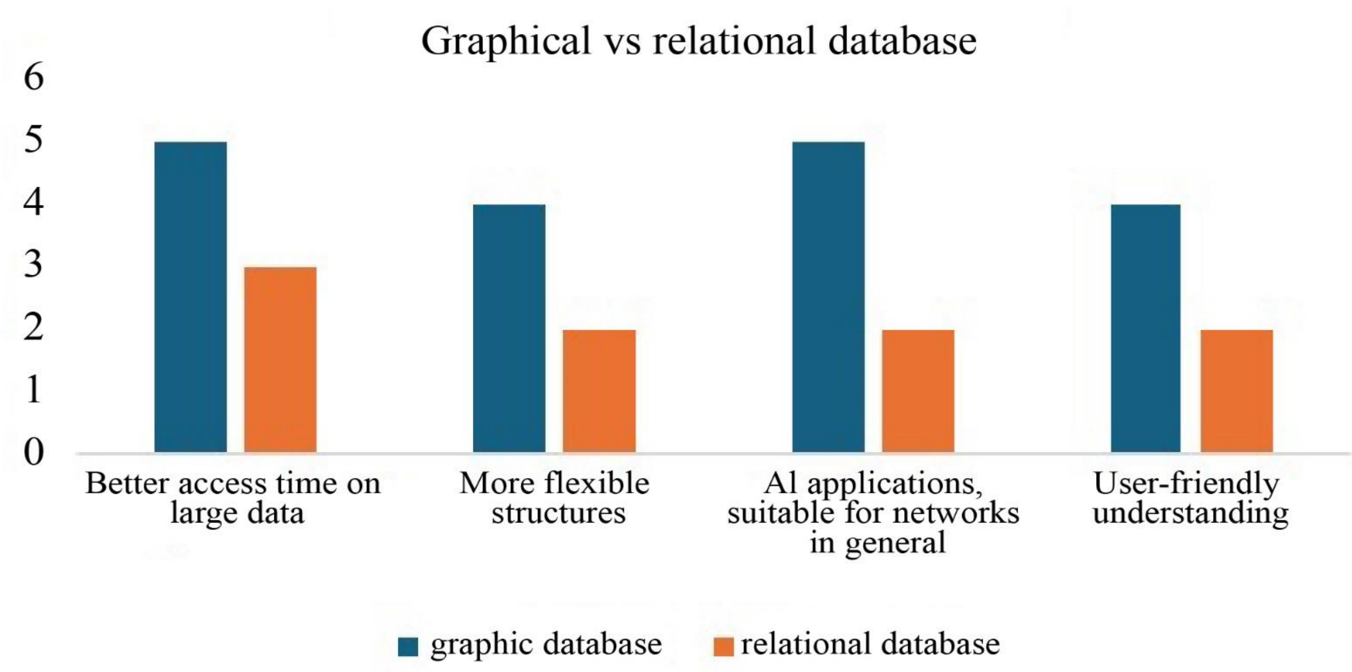

Fig. 2 Comparison of graphical vs. relational databases

Graph databases are gaining popularity as data grows in scale, and the importance of managing relationships surpasses the capabilities of relational databases.[15] Neo4j's graph algorithms offer highly efficient, parallel implementations of widely used graph algorithms, seamlessly integrated and specifically optimized for the Neo4j transactional database.[16] Neo4j demonstrated superior performance in terms of time and response compared to relational databases. It is particularly well-suited for handling complex networks of connections involving a large number of nodes and relationships.[7] Efficiency degrades in traditional databases when data grows exponentially, especially when a query is required, and several joints must be made. [17] Previous quotes read in state of the art, the comparison between relational databases and graphs in scala liker (1-5) is obtained in Figure 2, specifying that it is only when it comes to modeling suitable for applications in a graphical environment.

3. Materials and Methods

The modeling is based on graph theory, where a node is an entity, for example, a mining company, and the edges are the relationships, such as the extraction of the mineral, and the properties are the fields of the node, which could be the name of the mining company, its location, etc.; below is information from articles read regarding mining activity in Peru. Mining activity in Peru is concentrated in several regions of the country, each with its own characteristics and specializations. Ancash stands out for its importance in the production of copper; Arequipa is the main producer of copper, and the cities follow Apurimac, Cuzco and Junin. Regarding exports, China has become the main destination for Peruvian copper, reflecting the growing importance of Asian markets [18] Peru has exceeded global expectations for exporting minerals such as copper, silver and zinc, thus avoiding economic recession [19] The Peruvian mining sector has a diverse mix of national and international companies; some operate in multiple segments, and others specialize in specific minerals. Companies such as Buenaventura, which is listed on the New York Stock Exchange, are influenced by the prices of gold and silver, as these products hold significant weight in its export portfolio [20].

The diversification of export markets, which includes countries in Asia, Europe and America, is an important feature of the Peruvian mining sector, which helps mitigate the risks associated with dependence on a single market. [21] Faced with the challenge of structuring irregular data in the era of big data, graph databases offer an effective solution to manage relationships between data without the need for costly operations and provide great flexibility to adapt to changes in the data model.[22]. Examining relationships between data entities has resulted in their representation as graphs. With the rapid growth in the size of data sets in recent years, the need for efficient graph-based databases capable of loading and managing these massive data sets has become increasingly important [23].

4. Results and Discussion

4.1. Implementation in Neo4j

The implementation of the graph database in Neo4j was carried out following a structured and methodical process below at Node creation. We use Cypher's CREATE command to generate the parent nodes representing the key entities. Minerals. Types (Ferrous and non-Ferrous). Countries. Companies. Departments the key steps in this process.

Relationship ES_TIPO

```
// Relations ES_TIPO
MATCH (a:Minerales {name:'Hierro'})
MATCH (b:TIPOS {name: 'Ferroso'})
CREATE (a)-[:ES_TIPO]->(b)

MATCH (a:Minerales)
WHERE a.name IN
['Cobre', 'Zinc', 'Oro', 'Plomo']
MATCH (b:TIPOS {name: 'No Ferroso'})
    CREATE (a)-[:ES_TIPO]->(b)
```

This relationship connects the Mineral nodes with the TYPE nodes. Its function is to classify each mineral according to its metallurgical category, which can be seen in Figure 3, the types of minerals developed based on Cypher. Previous code generates this view.

Relationship ES_EXTRAIDO_EN

```
MATCH (a:Minerales {name:'Cobre'})
WITH a MATCH (b:Departamentos)
WHERE b.name IN
['Arequipa','Moquegua','Ancash','Tacna','Junin']
    CREATE (a)-[:ES_EXTRAIDO_EN]->(b)
```

This relationship links the Mineral nodes with the Department nodes. Its purpose is to indicate the geographic regions where each mineral is extracted, which can be seen in Figure 4. Previous code generates this view.

Relation EXTRAC

```
MATCH (a:Empresas {name:
'MineraBV'})
WITH a MATCH (b:Minerales)
WHERE b.name IN

['Oro', 'Cobre', 'Zinc']
CREATE (a)-[:EXTRAEN]->(b)
```

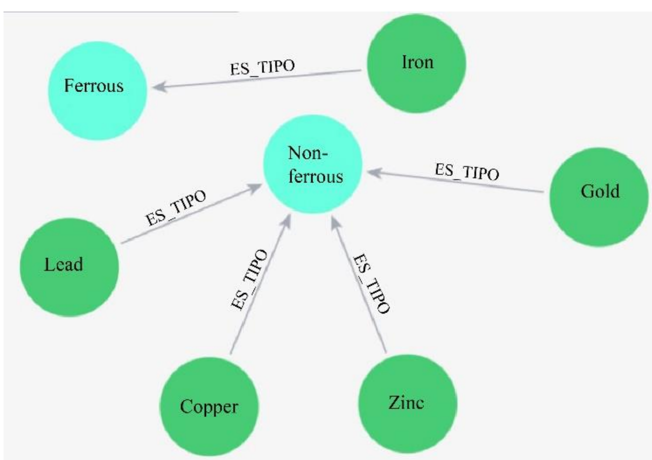

Fig. 3 Mineral Type

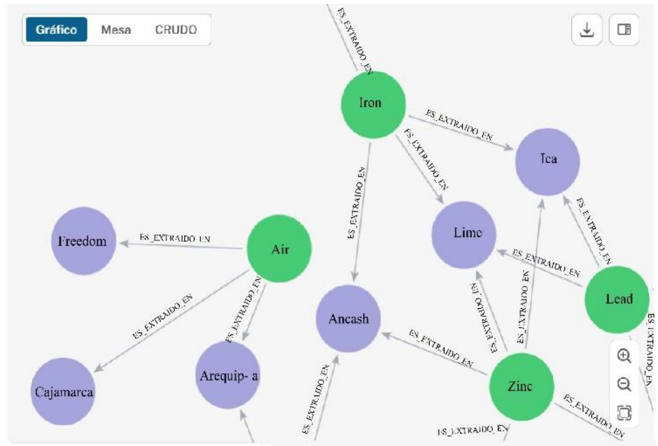

Fig. 4 Graphical model of the mineral extraction site relationship

Figure 5 shows the relationship between Enterprises and Minerals.

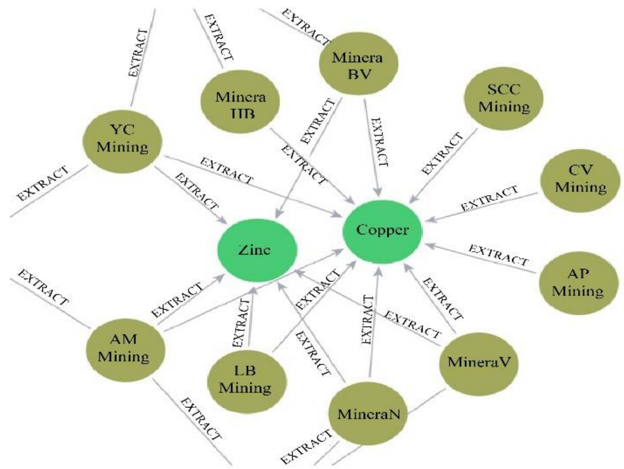

Fig. 5 Extraction of minerals by the mining company

This relationship links the company nodes with the minerals nodes. Its purpose is to show which mining companies are involved in the extraction of each mineral.

```
MATCH(a:Empresas{name:
'MINERACV'})
WITH a MATCH(b:Departamentos)
WHERE b.name IN ['Arequipa']

CREATE (a)-[:OPERA_EN]->(b)
```

Relation OPERA_EN

This relationship links the Company nodes with the Department nodes. Its purpose is to represent the geographic areas where each mining company has operations, as seen in Figure 6. Neo4j's flexible structure allowed us to capture the interconnected nature of the minerals, companies, departments and countries involved in the mining industry, facilitating deep analysis and extraction.

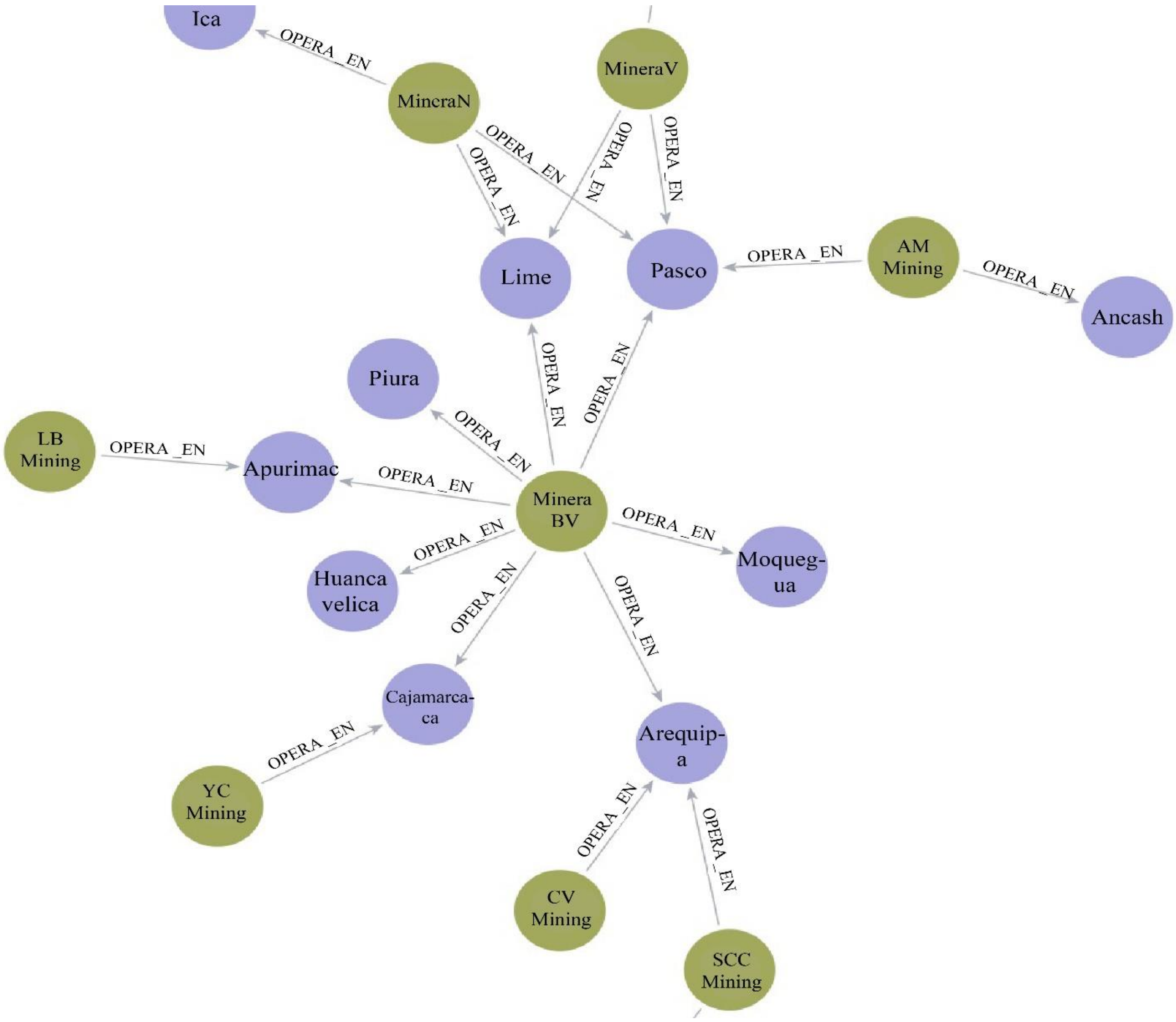

Fig. 6 Graphical model of the relationship between mines and the place where they operate

```
// Creación de nodos
CREATE(Cobre:Minerales {name: "Cobre"})
CREATE(Zinc:Minerales {name: "Zinc"})
CREATE(oro:Minerales {name: "oro"})
CREATE(Plomo:Minerales {name: "Plomo"})
CREATE(Hierro:Minerales {name: "Hierro"})
CREATE(Ferroso:TIPOS {name: "Ferroso"})
CREATE(No_Ferroso:TIPOS {name: "No Ferroso"})
// ... (creación de nodos para Países, Empresas y Departamentos)
// Creación de relaciones
MATCH (a:Minerales{name:'Hierro'})
MATCH (b:TIPOS{name: 'Ferroso'})
CREATE (a)-[:ES_TIPO]->(b)
// ... (creating ES_TIPO relationships for other minerals)
MATCH(a:Minerales{name:'Cobre'})
WITH a
MATCH(b:Departamentos)
WHERE b.name IN ['Arequipa','Moquegua','Ancash','Tacna','Junin']
CREATE (a)-[:ES_EXTRAIDO_EN]->(b)
// ... (creating relationships ES_EXTRAIDO_EN for other minerals)
```

```
MATCH(a:Minerales{name:'Cobre'})
WITH a
MATCH(b:Paises)
WHERE b.name IN ['China','Japón','Corea del Sur','Alemania','India']
CREATE (a)-[:SE_EXPORTA_A]->(b)
// ... (creating relationships SE_EXPORTA_A for other minerals)
MATCH(a:Empresas{name:'Compañía de Minas Buenaventura'})
WITH a
MATCH(b:Minerales)
WHERE b.name IN ['oro', 'Cobre', 'Zinc']
CREATE (a)-[:EXTRAEN]->(b)
// ... (building EXTRAEN relationships for other companies)
MATCH(a:Empresas{name:'Sociedad Minera Cerro Verde'})
WITH a
MATCH(b:Departamentos)
WHERE b.name IN ['Arequipa']
CREATE (a)-[:OPERA_EN]->(b)
// ... (creating OPERA_EN relationships for other companies)
// Finding all the minerals and their types
MATCH (m:Minerales)-[:ES_TIPO]->(t:TIPOS)
RETURN m.name AS mineral, t.name AS Tipo
// Finding the departments where copper is extracted
MATCH (m:Minerales {name: 'Cobre'})-[:ES_EXTRAIDO_EN]->(d:Departamentos)
RETURN m.name AS mineral, collect(d.name) AS Departamentos
// Finding the companies that mine Gold
MATCH (e:Empresas)-[:EXTRAEN]->(m:Minerales {name: 'Oro'})
RETURN m.name AS mineral, collect(e.name) AS Empresas
```

Following Figure 7, all the nodes and relationship

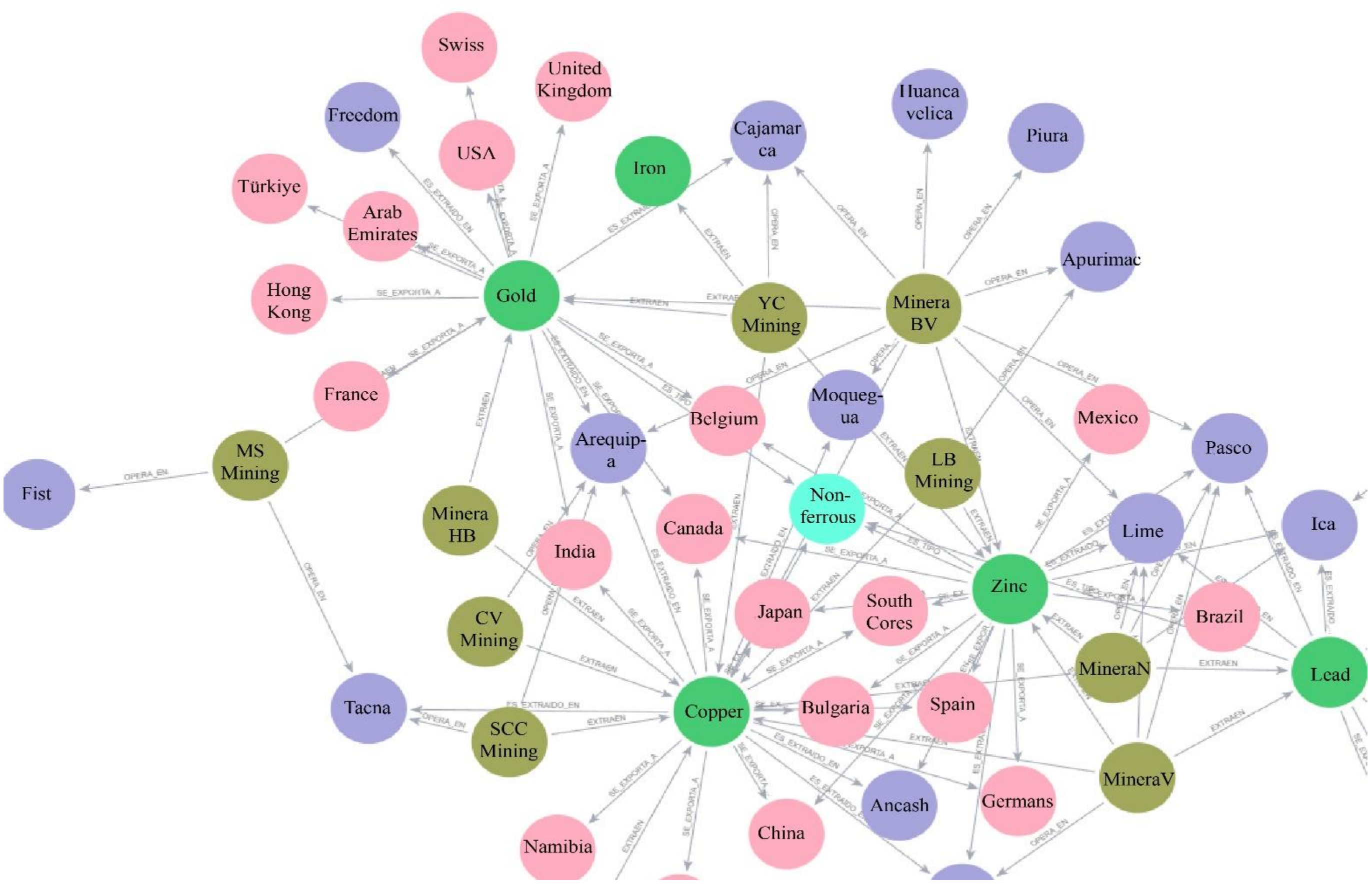

Fig. 7 General graphic model

4.2. Queries for Extracting Information

To validate the functionality and ability to extract relevant information, we designed and ran a series of test queries: Using Cypher queries, we were able to analyze the relationships between minerals, export countries, mining companies and departments. Some key findings include:

- **Mineral diversity:** The five main minerals analyzed are Copper, Zinc, Gold, Lead and Iron.
- **Classification of minerals:** Iron is classified as Ferrous, while Copper, Zinc, Gold and Lead are Non-Ferrous.

- **Mineral extraction**
    - Copper is the mineral most extracted by the various mining companies, this makes us understand that it is the most exported, generating great economic benefits for the nation.
    - Other minerals, such as zinc and gold, are also among the most extracted. Despite not having large extraction volumes, their market value and the various countries that buy them compensate for this difference in extraction compared to copper.
- **Concentration of exports**
    - China is the main export destination for Copper, Zinc and Iron. As can be seen in Figure 6, Switzerland is the main destination for Gold.
    - The following relationship is that of copper with the countries to which it is exported.

The Relationship SE_EXPORTA_A

```
MATCH (a:Minerales {name:'Cobre'})
WITH a MATCH (b:Paises)
WHERE b.name IN ['China','Japón','Corea del
Sur','Alemania','India']
CREATE (a)-[:SE_EXPORTA_A]->(b)
```

This relationship connects the Mineral nodes with the Country nodes. Its function is to represent the export destinations of each mineral.

```
MATCH(a:Empresas{name:'Minera Yanacocha'})
WITH a
MATCH(b:Minerales)
WHERE b.name IN ['oro', 'Cobre', 'Zinc', 'Hierro']
CREATE (a)-[:EXTRAEN]->(b)
```

Neo4j handles the table option as a complement to the graphs in Table 1.

Table 1. The main export country of each mineral

Mineral	Country
Copper (Cobre)	China
Gold (Oro)	Switzerland
Zinc	China
Iron (Hierro)	United States
Lead (Plomo)	China

Table 2. Main production department of each mineral

Mineral	Department
Copper	Ancash
Gold	Cajamarca
Zinc	Ancash
Iron	Pasco

Table 3. Main producing company of each mineral

Mineral	Company
Copper	MineraAM
Gold	Yanacocha
Zinc	MineraAM
Iron	MineraBV
Lead	MineraV

- **Geographic distribution of production**
    - Copper is extracted mainly in Arequipa, Moquegua, Ancash, Tacna and Junín.
    - Zinc is produced in Ancash, Junín, Cerro de Pasco, Lima and Ica.
    - Gold is extracted in La Libertad, Cajamarca and Arequipa.
    - Iron is produced in Ancash, Ica, Lima and Loreto.
    - Lead is extracted in Lima, Ica, Ayacucho and Cerro de Pasco.
      Table 2 shows the Mineral relationship Place from where it is extracted (Department of the Country)

- **Mining companies and diversification**
    - Companies such as MineraAM and MineraBV are involved in extracting multiple minerals.
    - Others, such as Shougang Hierro Perú and Minsur, specialize in a single mineral.
    - Most companies extract between 1 and 2 types of minerals. Table 3 shows the Mineral relationship and Mining Company.

- **Diversified exports**
    - Copper is mainly exported to Asian and European countries.
    - Zinc has a mix of Asian, European and South American destinations.
    - Gold has more diverse destinations, including Switzerland, India, Canada, the USA and the United Arab Emirates.
    - Lead is exported mainly to American countries, with the exception of Thailand.
    - Iron is mainly exported to Asian countries such as China and Japan. Figure 8 Shows the export relationship.

Table 4 shows the relationship of mineral, type of mineral and Department of the country (Perú). This graph-based analysis allows us to quickly visualize and understand the complex interrelationships in the Peruvian mining sector, facilitating the identification of trends that might not be evident in a traditional table-based analysis.

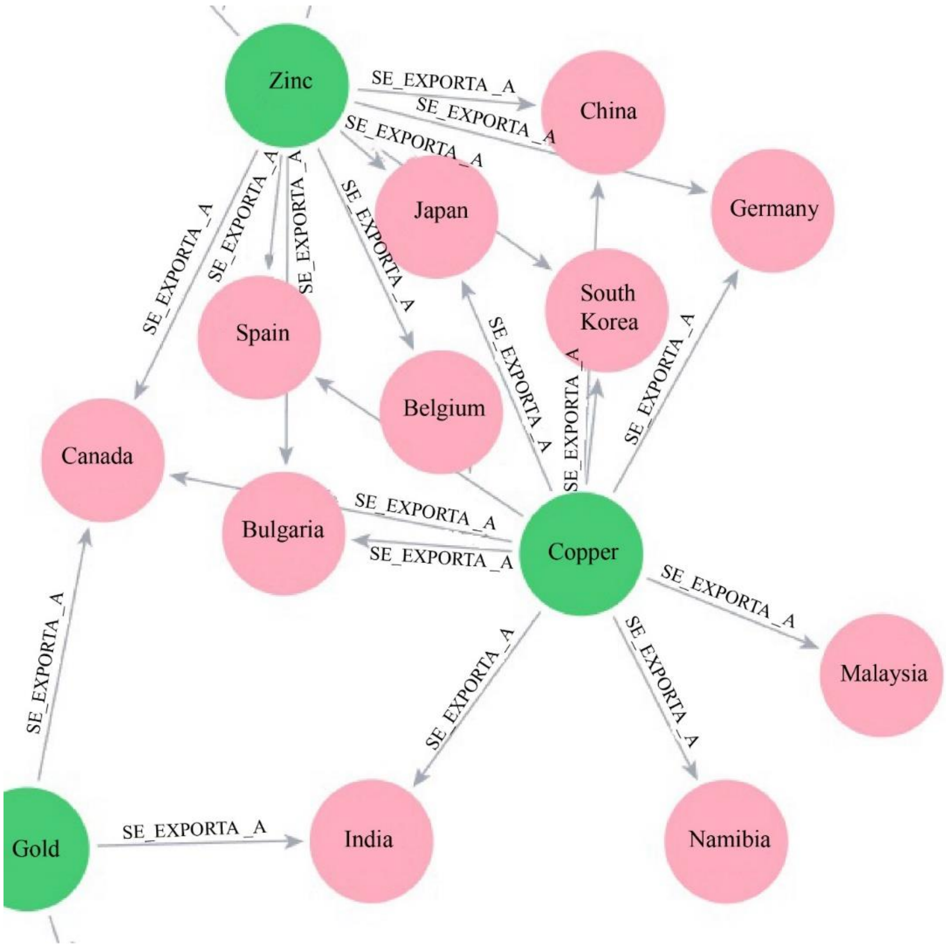

Fig. 8 Export relationship

Table 4. Peruvian minerals and their characteristics

Mineral	Type	Department
Copper	Non-ferrous	Ancash
Gold	Non-ferrous	Cajamarca
Zinc	Non-ferrous	Ancash
Iron	Ferrous	Pasco
Lead	Non-ferrous	Pasco

```
MATCH (m:Minerales {name: 'Cobre'})-
[:ES_EXTRAIDO_EN]->(d:Departamentos)
RETURN m.name AS mineral, collect(d.name) AS
Departamentos.
```

Although mining activity is important in the development of Perú, this sector implements social programs to avoid conflicts that harm the populations that are located in its surroundings, which is well received by the inhabitants, given the direct and indirect contributions to the respective municipalities, as is the case of the CANON MINERO.[24]. clarifying that mining canon is the percentage that mining companies pay, which is then transferred to the regions from where the mineral is extracted. This activity or product is defined in the financial world as commodities that generate high tax revenues. In response, the state promotes policies in favor of these activities. [25]

These activities have a high degree of impact on the GDP, contributing to the economic sustainability of the countries immersed in this activity.[26] However, mining activities present impacts and deterioration in health; therefore, the government of Perú has implemented a program to preserve the comprehensive health of the mining worker called "Occupational health of the worker", which includes frequent medical check-ups including his family. [27] Mining activity could undoubtedly support other activities, such as agriculture, financially. [28] It can also be seen in table mode, as seen in Figure 9 below.

The following code creates the distances from the Mine to the destination port through which it is exported, which is Callao.

```
CREATE (anta:City {name: 'MineraAM'}),
(ancash:City {name: 'Áncash'}),
    (lima:City {name: 'Lima'}),

   (huanuco:City {name: 'Huánuco'}),
```

```
(callao:City {name: 'Callao'})
// Create relationships with distances for routes
CREATE (anta)-[:CONNECTED {distance: 100}]->(ancash),
(ancash)-[:CONNECTED {distance: 200}]->(lima),
(lima)-[:CONNECTED {distance: 20}]->(callao),
(anta)-[:CONNECTED {distance: 150}]->(huanuco),
(huanuco)-[:CONNECTED {distance: 100}]->(lima),
(lima)-[:CONNECTED {distance: 70}]->(callao)
```

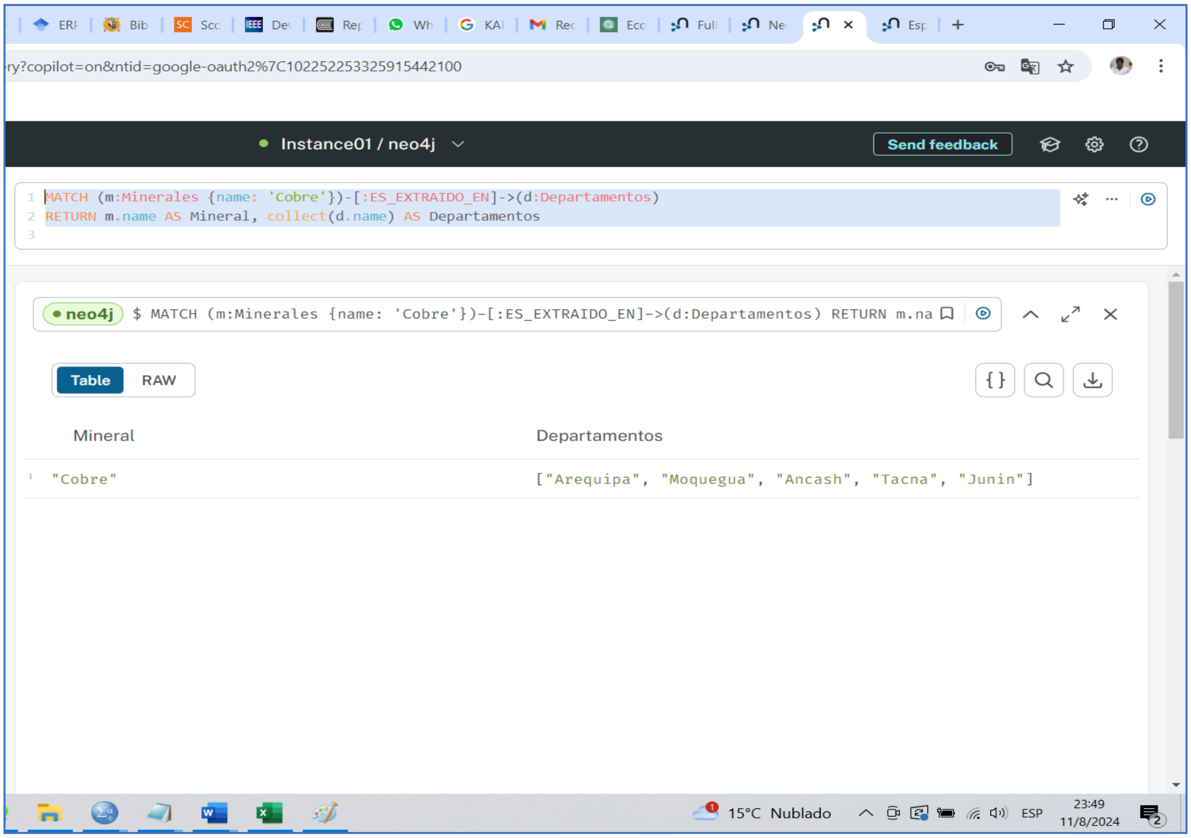

Fig. 9 List of mineral and extraction departments in table

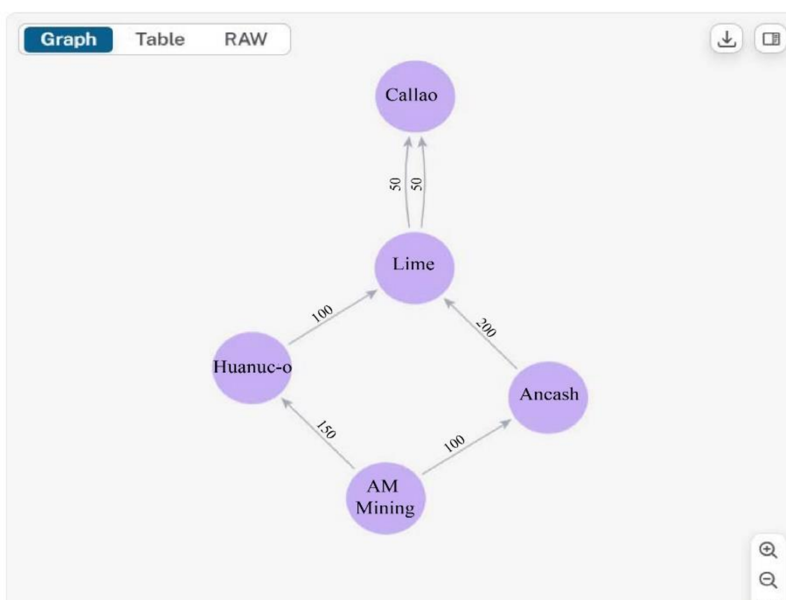

Fig. 10 MineraAM routes to the port of callao mineral shipping área

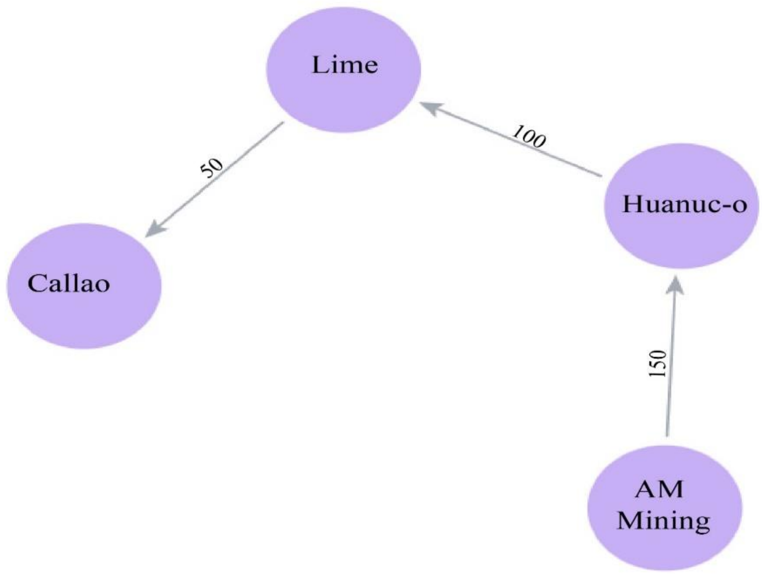

Fig. 11 Minimum route using dijkstra

Assigning the respective labels to the city and mining nodes.

```
// Asignar nombre a los nodos
MATCH (n)
WHERE id(n) = 52
SET n.name = ‘MineraAM'
RETURN n;
……
```

We can see in Figure 10 and the short path based on Dijkstra in Figure 11.

Using this code:

```
CALL gds.shortestPath.dijkstra.stream({
  nodeProjection: 'City',
  relationshipProjection: {
    CONNECTED: {

type: 'CONNECTED',
      properties: 'distance'
}
  },
  startNode: 'MineraAM',   // ID o nombre del nodo de
inicio
  endNode: 'Lima',        // ID o nombre del nodo de
destino
  relationshipWeightProperty: 'distance'
})
YIELD indice AS Indice,
nodoFuente AS NodoOrigen,
      nodoDestino AS NodoDestino,
      costoTotal AS CostoTotal,
      idsNodos AS IdsDeNodos,
      costos AS Costos,
      camino AS Camino
RETURN
  gds.util.asNode(NodoOrigen).name AS Origen,
  gds.util.asNode(NodoDestino).name AS Destino,
  CostoTotal,
  [idNodo IN IdsDeNodos |
gds.util.asNode(idNodo).name] AS
NombresNodosCamino
```

Dijkstra's algorithm deals with the shortest path problem in a weighted directed graph G=(V, E) G = (V, E), in which all edge weights are non-negative.[29] Comparing what was reviewed in the literature in section 2.4 in relation to the advantages over relational databases is that, as can be seen, it is easier for the user to visualize graphically the nodes and their relationships, apart from the queries.

They are faster, using less join. The results indicate that the graph database significantly surpasses the relational database when executing queries on large and complex datasets.[30]

5. Experiment

Pretesting for log queries, running on SQL Server vs Cypher. to consult the minerals produced by a specific Mining Company along with its information.

5.1. SQL Server

```
SET STATISTICS TIME ON;

SELECT mi.Nombre AS Mina,
       mn.Nombre AS Mineral,
       pr.Toneladas

FROM Mina mi
JOIN Produccion pr ON mi.Codigo = pr.CodigoMina
JOIN Mineral mn ON pr.CodigoMineral = mn.Codigo
```

```
WHERE mi.Nombre = 'Minera BV';

SET STATISTICS TIME OFF;
```

5.2. Cypher

```
PROFILE
MATCH (m:Minera {nombre: 'Minera BV'})-[:PRODUCE]->(min:Mineral)
RETURN m.nombre AS Mina, min.nombre AS Mineral, min.toneladas AS Toneladas;
```

With a large number of records, greater speed is observed in the graphic database, and inner joins delay access in relational databases; a deep query can be executed in milliseconds since the engine only needs to follow the pointers to the connected nodes.

6. Conclusion

In this article, descriptive research has been carried out for academic purposes, applying graph databases in the NEO4J environment in its free version relating it to the basic concepts of graph theory. Graph databases offer significant flexibility in representing and querying relationships. Unlike relational databases, where relationships must be managed through join tables and foreign keys, Neo4j relationships are a first class of citizens and can have properties and be queried directly. Graphical representation of data in Neo4j facilitates intuitive visualization of connections and relationships between data. This is particularly useful for exploratory analysis and understanding the structure of the data. Neo4j can solve complex problems related to roads and routes efficiently, which is useful for applications in logistics, social networks, and route analysis. The Cypher query language is intuitive and allows complex queries to be expressed clearly and concisely. This makes it easier to query and manipulate data In summary, Neo4j and the Cypher language provide powerful tools for working with data that has a complex relationship structure. The ability to naturally model and query relationships, along with efficiency in solving path and route problems, makes Neo4j an ideal choice for applications that require deep analysis of connectivity and distance between data.

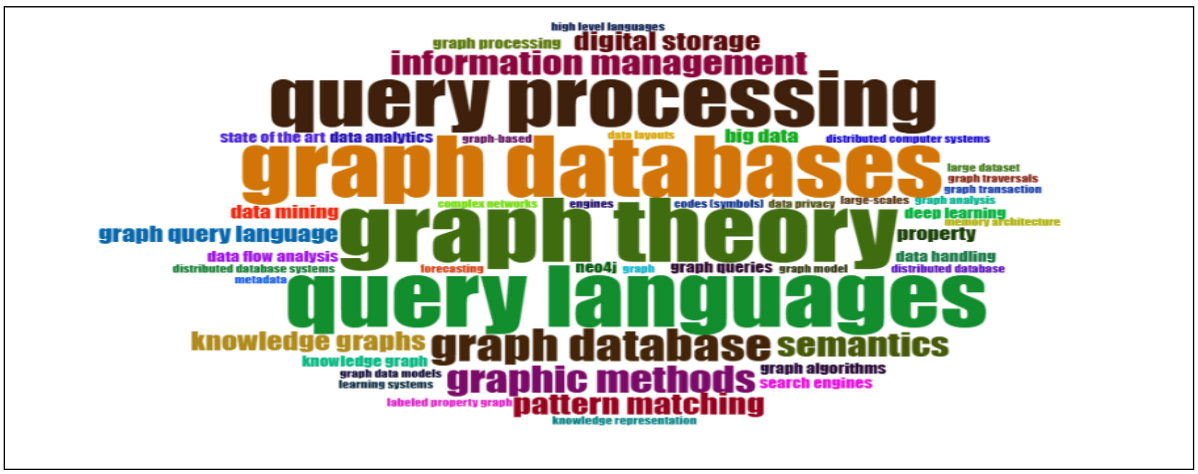

Fig. 12 Trends in graph theory

Cypher's flexibility in relationship representation and ease of use are additional advantages that contribute to the usefulness of graph databases compared to traditional relational databases. Graph databases are based on graph theory, an area of mathematics that studies the relationships between entities through nodes (entities) and edges (relationships). These databases are designed to manage and query highly connected data efficiently, overcoming the limitations of traditional relational databases in complex queries. Finally, it can be seen that it is suitable for the application referring to mining companies. Nowadays, in graphical applications for artificial intelligence purposes, there are a series of algorithms, including Graphical Neural Networks (GNNs), which capture the complex relationships and structures of the data that are in the nodes (entities) and edges (relationships) by interweaving the various nodes in order to make predictions. In addition, graph theory has wide applications, and it can be seen in bibliometric analysis based on the articles read according to the search string used, which can be seen in Figure 12.

References

[1] Gerson Enrique Carrillo Gelvezac, and Ixent Galpinbd, "Graph Analytics to Identify Relevant Entities and Communities in Mercado Libre: A Case Study," *Mutis Magazin*, vol. 11, no. 1, pp. 77-95, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Santo Fortunato, "Community Detection in Graphs," *Phisics Report*, vol. 486, no. 3-5, pp. 75-174, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[3] Jonathan D. Gonzalez-Barrera, and Francisco J. Rodríguez y Manuel Lozano, "Iterative Greedy Algorithms for the Maximum Diversity Clustering Problem," *VIII Spanish Conference on Metaheuristics, Evolutionary and Bioinspired Algorithms MAEB'2012*, pp. 1-8, 2012. [Google Scholar] [Publisher Link]

[4] Cristina Pérez-Solà, Jordi Conesa i Caralt, and Rodríguez González M. Elena, *How to use a Graph Database?*, Torrossa, Barcelona:Editorial UOC Publishing, pp. 1-90, 2017. [Google Scholar] [Publisher Link]

[5] Renzo Angles, and Claudio Gutierrez, "Survey of Graph Database Models," *ACM Computing Surveys (CSUR)*, vol. 40, no. 1, pp. 1-39, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[6] Ian Robinson, Jim Webber, and Emil Eifrem, *Graph Databases: New Opportunities for Connected Data*, O'Reilly Media, 2015. [Google Scholar] [Publisher Link] .

[7] T.P. Deshmukh, B.R. Bamnote, and S.W. Ahmad, "Review Paper On Complexity Reduction in Relational Database Using Neo4j Graph Database," *International Journal of Advances in Engineering and Management*, vol. 2, no. 9, pp. 603-610, 2020. [Google Scholar] [Publisher Link]

[8] Mohamed Saad et al., "A Graph Database for Life Cycle Inventory using Neo4j," *Journal of Cleaner Production*, vol. 393, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[9] Jose Guia, Valeria Gonçalves Soares, and Jorge Bernardino, "Graph Databases: Neo4j Analysis," *Proceedings of the 19th International Conference on Enterprise Information Systems (ICEIS 2017)*, Porto, Portugal, vol. 1, pp. 351-356, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[10] Maciej Besta et al., "Demystifying Graph Databases: Analysis and Taxonomy of Data Organization, System Designs, and Graph Queries," *ACM Computing Surveys*, vol. 56, no. 2, pp. 1-40, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Qinjun Qiu et al., "Information Extraction and Knowledge Linkage of Geological Profiles and Related Contextual Texts from Mineral Exploration Reports for Geological Knowledge Graphs Construction," *Ore Geology Reviews*, vol. 163, pp. 1-14, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Malgorzata Lazarska, and Olga Siedlecka- Lamch, "Comparative Study of Relational and Graph Databases," *2019 IEEE 15th International Scientific Conference on Informatics*, Poprad, Slovakia, pp. 363-370, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[13] Garima Jaiswal, and Arun Prakash Agrawal, "Comparative Analysis of Relational and Graph Databases," *IOSR Journal of Engineering (IOSRJEN)*, vol. 3, no. 8, pp. 25-27, 2013. [Google Scholar] [Publisher Link]

[14] Jaroslav Pokorný, Michal Valenta, and Martin Troup, "Graph Pattern Index for Neo4j Graph Databases," *Data Management Technologies and Applications*, pp. 69-90, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[15] Diogo Fernandes, and Jorge Bernardino, "Graph Databases Comparison: AllegroGraph, ArangoDB, InfiniteGraph, Neo4J, and OrientDB," *Proceedings of the 7th International Conference on Data Science, Technology and Applications (DATA 2018)*, Porto, Portugal, vol. 1, pp. 373-380, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[16] David Allen et al., *Understanding Trolls with Efficient Analytics of Large Graphs in Neo4j*, Gesellschaft für Informatik, Bonn, pp. 377-396, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Xianyu Meng, Xiaoyan Cai, and Yanping Cui, "Analysis and Introduction of Graph Database," *3rd International Conference on Artificial Intelligence and Advanced Manufacture*, Manchester United Kingdom, pp. 392-396, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Edelina Coayla, Violeta L. Romero Carrion, and Ysabel Teófila Bedón Soria, "Economic Regulation and Environmental Impact of Large Copper Mining in the Development of Peru," *Economy, Society and Territory*, vol. 24, no. 74, pp. 1-24, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[19] Ketty M. Moscoso-Paucarchuco, Manuel M. Beraún-Espíritu, and Mary A. Cárdenas-Bustamante, "Contribution of Mining in Terms of Canon and Other Royalties in Peru," *Valdizana Research*, vol. 15, no. 2, pp. 57-68, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Nicko Alberto Gomero Gonzales, "Transversal Influence of Mining Commodities on the Peruvian Economy Under a Global Approach," *Quipukamayoc*, vol. 29, no. 59, pp. 25-34, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] L. Alfredo Dammert, "The Importance of the Mining Sector for Peru," *Journal of Economics, Finance and International Business*, vol. 4, no. 1, pp. 1-7, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[22] Jinhua Chen et al., "Graph Database and Relational Database Performance Comparison on a Transportation Network," *Advances in Computing and Data Sciences, Communications in Computer and Information Science*, vol. 1244, pp. 407-418, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[23] D. Dominguez-Sal et al., "Survey of Graph Database Performance on the HPC Scalable Graph Analysis Benchmark," *Web-Age Information Management*, pp. 37-48, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[24] Wilwer Vilca-Quispe et al., "Mining Corporate Social Responsibility and Socio-Environmental Conflicts in Peru," *International Journal of Social Science Research*, vol. 17, no. 1, pp. 195-215, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Viviana Soledad Verastegui Pilcon et al., "Income Tax and Profitability in the Peruvian Mining Sector from 2017 to 2021," *LATAM Latin American Journal of Social Sciences and Humanities*, vol. 4, no. 1, pp. 3768-3777, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[26] Ricardo Viana Rios, "Mining in Latin America and the Caribbean, A Socio-Environmental Approach," *UDCA Journal Current Affairs & Scientific Dissemination*, vol. 21, no. 2, pp. 617-631, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[27] Christian Renzo Aquino-Canchari, Katia Medalith Huaman-Castillon, and Fatima Jimenez-Mozo, "Occupational Diseases in Mining in Peru, 2011-2020," *Journal of the Spanish Association of Occupational Medicine Specialists*, vol. 31, no. 3, pp. 275-282. 2022. [Google Scholar] [Publisher Link]

[28] Oscar Junior Paredes-Vilca et al., "Pollution and Loss of Biodiversity Due to Mining and Agricultural Activities: State of The Art," *High Andean Research Journals*, vol. 26, no. 1, pp. 56-66, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[29] Thomas H. Cormen et al., *Introduction to Algorithms*, United States of America, MIT Press and McGraw-Hill, pp. 1-1312, 2001. [Google Scholar] [Publisher Link]

[30] Mit Jain, Ashish Khanchandani, and Cajetan Rodrigues, "*Performance Comparison of Graph Database and Relational Database*," pp. 1-13, 2023. [Google Scholar]