International Journal of Computer Trends and Technology
ISSN: 2231–2803 / https://doi.org/10.14445/22312803/IJCTT-V71I10P107
Volume 71 Issue 10, 51-60, October 2023

Original Article

Advancing Financial Operations: Leveraging Knowledge Graph for Innovation

Shyam Balagurumurthy Viswanathan[1], Gaurav Singh[2]

[1]*City University of New York, GA, USA.*
[2]*Cyber Security Manager, Maryland, USA.*

Received: 23 August 2023 Revised: 26 September 2023 Accepted: 08 October 2023 Published: 26 October 2023

Abstract *- Recently, there has been a growing interest in Knowledge Graphs (KG) due to their ability to systematically structure and categorize complex information. These graphical representations uniquely identify intricate patterns, reveal hidden insights, process enormous amounts of data with intuitive visualization, and accurately retain diverse information. One domain where KGs can be particularly valuable is the financial market, specifically the stock sector, which generates vast amounts of data across various platforms. This research paper outlines the methodologies for constructing a knowledge graph using graph databases and explores how it can enhance our understanding of financial paradigms. By doing so, this paper aims to improve the accuracy and depth of stock analyses, facilitate better decision-making processes, and detect anomalies such as intellectual property theft or insider trading within the realm of finance.*

Keywords *- Knowledge graph, Financial operations, Cyber security, Trading, Intelligence.*

1. Introduction

Knowledge graphs have become popular in various domains, such as recommendation systems, software design and development, market prediction, cybersecurity, IP, and stock investments [1]. One example of using knowledge graphs is music recommendation. Oramas et al. [2] used knowledge graphs to recommend sounds and music. Another example is the integration of knowledge graphs with SQL databases proposed by Li et al. [3], demonstrating the effectiveness of using knowledge graphs in industrial software design and development processes. In addition, Liu et al. [4] employed a combination of deep learning and knowledge graph integration to prove the efficacy of market prediction and stock investments. Knowledge graphs provide a means to organize and analyze data, making them an appealing tool for predicting stock market trends. In the stock market, a wealth of data is available, including historical prices, company news, insider trades, and critical metrics about companies. However, this information is often scattered and disjoined, making it difficult to extract meaningful insights. To address this challenge, using knowledge graphs can be highly beneficial.

2. Literature Review

In the ever-evolving digital era, the financial industry, particularly the stock market, grapples with a vast influx of data from various sources such as news outlets, financial reports, and social media. Effectively managing this extensive information necessitates tools capable of not only organizing it but also extracting valuable insights.

Knowledge graphs have emerged as a promising solution rooted in semantic web technologies. These graphs provide a structured representation of data that seamlessly connects disparate pieces of information. This structured representation becomes invaluable in sectors like finance, where understanding entity relationships is critical for making informed decisions.

2.1. Knowledge Graphs

Knowledge graphs are understood as networked datasets that associate entities with their attributes and interrelationships. Their utility has been recognized across various sectors, from e-commerce to healthcare, exemplifying their versatility.

2.2. Knowledge Representation in Finance

The financial domain, with its complex interplay between assets, companies, and market indicators, presents a compelling case for knowledge graph implementations. Through KGs, latent connections, often obscured in traditional data processing approaches, can be unveiled.

2.3. Graph Databases in Finance

Research into the utility of graph databases for financial data storage and retrieval suggests that these databases facilitate efficient data access and enable advanced analytics. This makes them suitable for constructing knowledge graphs in the financial realm.

2.4. Stock Market Analysis

Conventional methodologies for analyzing stock market dynamics often overlook the interconnected nature of financial entities. This gap underscores the value of knowledge graphs as a potential addition to analytical toolkits in finance.

2.5. Insider Trading and IP Theft Detection

Traditional strategies for detecting market malpractices rely on surface-level data analysis. However, the depth and breadth of data representation provided by knowledge graphs can uncover patterns that might be missed by conventional methods, positioning them as crucial tools for market surveillance.

Traditional methods are commonly used for analyzing financial data, but there is limited utilization of graph databases in the context of financial stock markets. This leads to loosely integrated datasets and poses challenges when combining new datasets within a cloud graph database to handle the vast amount of information found in financial markets. This study aims to tackle these issues by efficiently integrating diverse datasets within a scalable cloud graph database designed specifically for financial markets.

3. Understanding Stocks in New Ways

Before building a knowledge graph for stocks [5], it is essential to understand what stocks are and how they function in the financial market. A stock is a type of security that symbolizes an ownership position in a corporation and represents a claim on the part of the company's assets and earnings. For example, a company's stock might trade at $ 90 per share on an exchange. You might be wondering what could be done with this data. In the past, conventional stock price prediction methods predominantly relied on time-series and regression analyses.

While traditional models assume linear relationships between variables, they might not be able to capture the complexity and the non-linear nature of financial markets. More recently, as AI and machine learning with graph algorithms continue to thrive and large and varied data becomes widely available, a new and promising approach that relies on these algorithms has been adopted. The input a variety of information available about financial instruments, and it can incorporate all the knowledge and derive insights from it. It can then be used to predict stock market trends and potentially make informed investment decisions and indirect relationships between organizations.

4. Variety of Data Sources

A knowledge base refers to a collection of information that establishes connections between entities and objects, facilitating the formation of relationships. Knowledge graphs leverage semantic associations to provide contextual understanding and facilitate integration, unification, analysis, and data sharing. It is highly recommended to explore further literature on knowledge graphs. From the context of the stock market:

- Entities mentioned in news articles contribute valuable insights.
- Daily news updates are released concerning specific stocks.
- Information from platforms like Reddit [6] regarding meme stocks can be significant.
- Parsing 10K-10Q [7] filings provides access to crucial data points.
- Critical metrics pertaining to companies play an instrumental role.
- Moreover, there exists a plethora of other relevant information.

One work relevant to the concept of this paper is done by Deng et al. [8], where a knowledge-driven event embedding model is developed, and the results indicate that financial market prices help forecast future financial market prices.

Despite their dispersed nature, these discrete pieces of information hold immense value as they connect disparate components within the domain under consideration. Given this and the data mentioned above [1], analyzing the stock market by forming knowledge graphs is beneficial, enabling investors to understand the underlying dynamics of stocks, businesses, and the industry in general.

Furthermore, the integration of this data can create a comprehensive knowledge graph. The practical applications of such a graph are manifold and extensive. It can facilitate efficient financial decision-making based on accurate information and provide invaluable insights for predicting stock market trends. This research places emphasis on harnessing the power of knowledge graphs to enhance the precision of stock price forecasts through meticulous analysis and evaluation methods.

- Graph neural networks can be formed.
- It could be a recommendation on stocks to buy.
- Follow patterns of institutional investment.
- Find a similar pattern of different stocks.
- Understand a stock from a bird's eye view.

5. Linking Knowledge Graphs and Stock Market Graph Database

Multiple graph databases exist in the market, offering many options for constructing knowledge graphs. AWS Neptune was selected as the primary graph database platform to reduce setup time and begin the research promptly. It is suggested to delve into its features when thinking about crafting a custom knowledge graph solution. Alternatives include Janusgraph [10], neo4j [11], Apache TinkerGraph [12], and tiger-graph [13]. A predominant hurdle when working with graphs is discerning which query language to employ. Conventional SQL knowledge is not directly translatable for querying graph data.

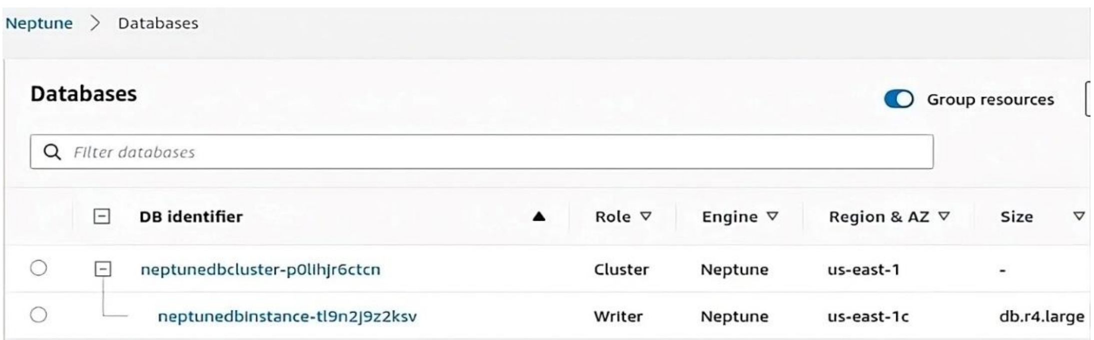
Various query languages exist, such as Gremlin [12], Cypher [14], SPARQL [15], and more, each presenting its own set of challenges. This paper will focus on Gremlin from Apache as an open-source choice. Although its learning process is straightforward, there are limitations to bear in mind.

6. Coding Language and Libraries

To access APIs and retrieve data, Python is commonly used for coding. One such API that provides all the necessary information in a packaged format is financial modeling prep [16]. Additionally, you will need the Gremlin Python package to load and work on graphs pip install Gremlin Python.

6.1. Graph Setup

To create the graph, this paper will use the Gremlin Python package and native Gremlin queries. Begin by focusing on a single stock, AAPL [17]. Connect to the AWS Neptune database from the AWS.



Neptune > Databases

Databases

Group resources

Filter databases

DB identifier	Role	Engine	Region & AZ	Size
neptunedbcluster-p0lihjr6ctcn	Cluster	Neptune	us-east-1	-
neptunedbinstance-tl9n2j9z2ksv	Writer	Neptune	us-east-1c	db.r4.large

Fig. 1

```python
from gremlin_python import statics
from gremlin_python.process.anonymous_traversal import traversal
from gremlin_python.driver.driver_remote_connection import DriverRemoteConnection
from gremlin_python.structure.graph import Graph

from pprint import pprint


target = 'neptunedbinstance-xxxxxxx.xxxxxxx.us-east-1.neptune.amazonaws.com'
port = '8182'

g = Graph()

remote = DriverRemoteConnection(f'wss://{target}:{port}/gremlin', 'g')
conn = g.traversal().withRemote(remote)
```

Fig. 2

6.2. Fetch Data

As previously noted, abundant data can be linked to the stock. Nonetheless, the primary objective should be to connect the stock's institutional investors. To facilitate this, enumerating all the NASDAQ [18]-related stocks is essential.

```python
import requests
from credentials import KEY
nasdaq = requests.get(f’https://financialmodelingprep.com/api/v3/nasdaq_constituent?apikey={KEY}’).json()
holdings = requests.get(f’https://financialmodelingprep.com/api/v3/institutional-holder/AAPL?apikey={KEY}’).json()
```

```
nasdaq
```


```
[{'symbol': 'DOCU',
  'name': 'DocuSign Inc',
  'sector': 'Technology',
  'subSector': 'Technology',
  'headQuarter': 'San Francisco, CALIFORNIA',
  'dateFirstAdded': '2020-06-22',
  'cik': '0001261333',
  'founded': '2018-04-27'},
 {'symbol': 'ZM',
  'name': 'Zoom Video Communications Inc',
  'sector': 'Communication Services',
  'subSector': 'Communication Services',
  'headQuarter': 'San Jose, CALIFORNIA',
  'dateFirstAdded': '2020-04-30',
  'cik': '0001585521',
  'founded': '2019-04-18'},
 {'symbol': 'DXCM',
  'name': 'DexCom Inc',
  'sector': 'Healthcare',
```


Fig. 3

```
holdings
```


```
[{'holder': 'Johnson Bixby & Associates, LLC',
  'shares': 43635,
  'dateReported': '2021-06-30',
  'change': 43635},
 {'holder': 'Kanen Wealth Management LLC',
  'shares': 7080,
  'dateReported': '2021-06-30',
  'change': 7080},
 {'holder': 'EP Wealth Advisors, LLC',
  'shares': 1097597,
  'dateReported': '2021-06-30',
  'change': 1097597},
 {'holder': 'Strategic Wealth Partners, Ltd.',
  'shares': 54896,
  'dateReported': '2021-06-30',
  'change': 54896},
 {'holder': 'Accurate Wealth Management, LLC',
  'shares': 49680,
  'dateReported': '2021-06-30',
```


Fig. 4

6.3. Create Graph

The ticker AAPL was filtered and incorporated as a vertex in the graph to optimize the procedure. All institutional holdings of AAPL were also included. Rather than adding edges in the prior code, the method highlighted involves using gremlin queries to establish connections by appending them to a list. Parsing the edge ID becomes necessary when crafting Gremlin queries with the ID.

```
nasdaq_appl = [item for item in nasdaq if item['symbol'] =='AAPL']
```

```
connection =[]

for ids,item in enumerate(nasdaq_appl,1):
    v1 = conn.addV('ticker').property('name', item['symbol']).next()

    for ids,item in enumerate(holdings,1):
        v2 = conn.addV('holder').property('name', item['holder']).next()

        connection.append((v1,v2))
```

Fig. 5

```
for item in connection[:10]:
    print("g.V('{}').addE('holdings').to(g.V('{}')).next()" \
         .format(str(item[0]).split('[')[1].split(']')[0],str(item[1]).split('[')[1].split(']')[0]))
```

Fig. 6

```
%%gremlin
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('4abdc754-6875-e9b7-6d96-ae4c07028ec0')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('16bdc754-687f-2837-682e-09df5add6dad')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('dabdc754-6886-9f49-a0fa-4e754e5fce54')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('38bdc754-688e-115a-5a4e-e6428bb7fd29')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('aabdc754-6896-1ac5-0ee4-3b81642fdeb6')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('30bdc754-6899-cff2-75ea-1811ff79a93b')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('dcbdc754-689e-e3a2-756b-993f582b8692')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('26bdc754-68a2-62fc-c1e2-b45e8646ad8a')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('06bdc754-68a5-959e-e22e-b75c0d4f8978')).next()
g.V('debdc754-6869-b973-9489-d9dfae7becc4').addE('holdings').to(g.V('d4bdc754-68a9-18c6-3a3b-aa480c8bf7a0')).next()
```

Fig. 7

6.4. Creating Edges – Holdings

Upon executing the required commands, connections between vertices can be successfully established. The completed graph for AAPL appears below, presented in a simplified manner that omits details about top investors for the sake of clarity and simplicity.

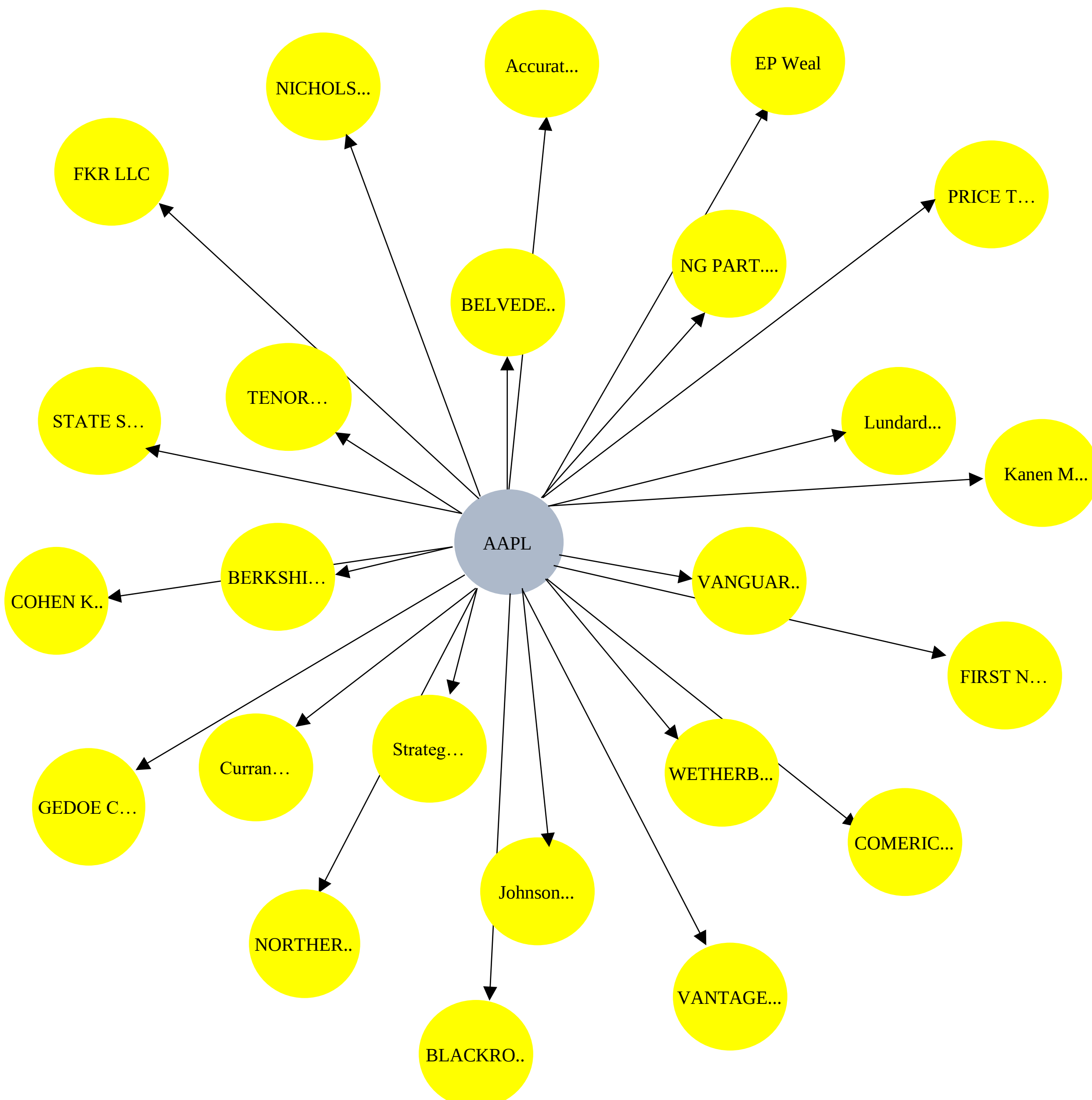


Fig. 8

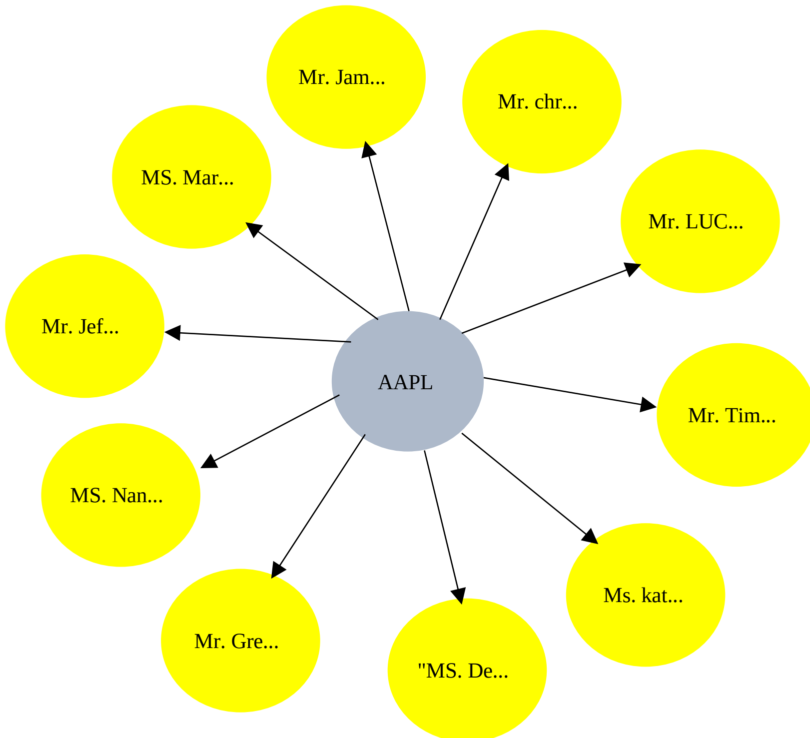


Fig. 9

6.5. Creating Edges – Executives

Financial model prep also provides data about the company leadership. Like the above code, we can create vertex and edges for AAPL.

6.6. Creating Edges - News Entities

Constructing entities from news requires intricate processing. Some basic processing has been undertaken, but more complex methods are available. Initially, gathering all the news and their summaries is suggested.

The preferred choice is the summary of each article. Selecting the article's summary aids in expediting the processing time. Although the entire article could be analyzed for additional entities, once the summary is acquired, it is imperative to establish a transformer NER pipeline to identify the entities. Numerous methods exist for entity extraction, and the optimal path should be determined.

The subsequent code remains consistent, which involves the addition of the identified entities to the ticker. The subsequent image can visualize the relationship between the entity and the ticker. Some extraneous entities might still be visible, which can be sifted out, but it is advised to continue forward.

```
nlp_ner = pipeline("ner", grouped_entities= True)

all_entities = []

for item in news:
    ner = nlp_ner(item['text'])

    entities = [item['word'] for item in ner if '#' not in item['word']]

    all_entities.append(entities)

all_entites_flat = list(set(pydash.flatten(all_entities)))
```

Fig. 10

By combining all the nodes, the comprehensive knowledge graph containing the collated information will be unveiled.

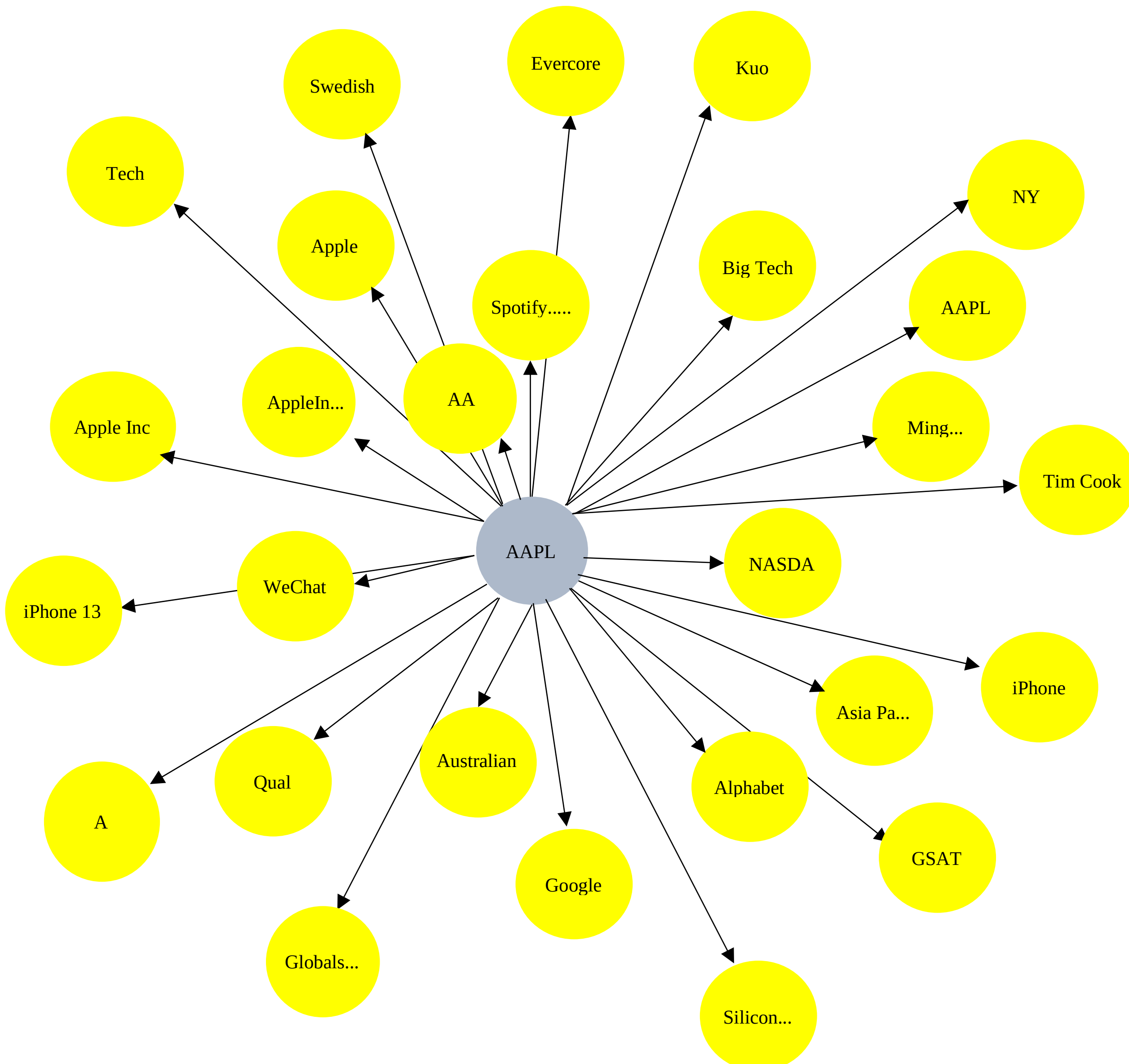


Fig. 11

7. Methodology

A knowledge graph has been successfully developed using AWS Neptune for a specific stock, with the capability to continually incorporate additional stocks.

Upon integrating subsequent stock network graphs, verifying the presence of existing nodes is advisable. For instance, identifying the affiliations of leadership roles and associating them with the appropriate vertex is recommended.

- It is crucial to append labels, tags, and relevant information to the vertex.
- It is beneficial to utilize previously established vertices for entities. In conjunction with AWS Neptune, knowledge graphs exhibit robust scalability, offering substantial data storage capacity.

Nevertheless, despite the inherent scalability and storage competencies of knowledge graphs and AWS Neptune [19], various challenges manifest in the development of stock knowledge graphs. A primary concern is acquiring and refining the extensive data inherent in the stock market, encompassing news articles, summaries, insider trades, financial disclosures, and other salient data points. Frequently, a singular API might be insufficient, necessitating interfacing with multiple APIs and subscription services to aggregate comprehensive data. Furthermore, securing permissions to utilize specific data outcomes could be imperative.

Rigorous evaluation is indispensable when selecting models for Named Entity Recognition. A judiciously curated

NER pipeline [20] can be employed to derive entities from designated summaries and articles. Although a plethora of models exist for such tasks, it remains paramount to assess considerations such as data volume and requisite processing power. Large Language Models may be harnessed in certain scenarios to ensure precise NER determinations. After entity extraction, their incorporation as nodes in the knowledge graph is the recommended course of action.

Fig. 12

8. Future Research

Applying knowledge graphs in the stock market provides many benefits and possibilities for further investigation and analysis. Graphs could be used to find out the hidden relationships. For example, the Board of Directors of two companies can be found, and the intellectual information shared between them.

- Find the companies that are so close.
- Create a Graph Neural Network using DGL [21] or PyTorch-geometric. [22]

- A ton of insights can be created.
- Another major item that can be created is a recommendation engine for stocks. It can be built depending on the portfolio.

9. Conclusion

This research article undertakes a thorough investigation into the application of knowledge graphs in the financial sector, specifically focusing on their role in the stock market. It highlights the significant value and utility of these graphs by providing detailed insights into constructing and extracting essential entities, summaries, and other relevant information. This work not only contributes to academia but also offers practical guidance for practitioners looking to implement effective strategies.

The research endeavors to incorporate the gathered information into AWS Neptune, emphasizing the need for a dependable platform and highlighting the process of constructing a knowledge graph from its inception. Recognizing the ever-changing nature of data in contemporary times, there is an emphasis on regularly updating and expanding datasets. This ongoing integration guarantees that the knowledge graph remains exhaustive and accurately reflects the complex intricacies of the financial domain.

In summary, this article emphasizes the importance of thorough analysis following the development of a knowledge graph. These analytical steps are crucial in ensuring that the created graph effectively serves its role as a valuable tool for making informed financial decisions. The rapidly changing global financial landscape demonstrates that knowledge graphs play a significant role within the stock market. This research not only highlights their significance but also lays the groundwork for future studies and applications in harnessing their potential benefits.

References

[1] Zhibo Li et al., “Bearing Fault Diagnosis Method Based on Convolutional Neural Network and Knowledge Graph,” *Entropy*, vol. 24, no. 11, pp. 1-18, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[2] Sergio Oramas et al., “Sound-and-Music-Recommendation-with-Knowledge-Graphs,” *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 2, pp. 1-21, 2016. [CrossRef] [Google Scholar] [Publisher Link]
[3] Wen-Dong Li et al., “Is Being a Leader a Mixed Blessing? A Dual-Pathway Model Linking Leadership Role Occupancy to Well-Being,” *Journal of Organizational Behavior*, vol. 39, no. 8, pp. 971-989, 2018. [CrossRef] [Google Scholar] [Publisher Link]
[4] Mingfei Liu et al., “A Knowledge Graph-Based Approach for Assembly Sequence Recommendations for Wind Turbines,” *Machines*, vol. 11, no. 10, pp. 1-25, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[5] B.V. Shyam, How to Build a Knowledge Graph on Stocks, AWS Neptune, 2021. [Online]. Available: https://medium.com/code-sprout/build-a-knowledge-graph-on-stocks-aws-neptune-7c2b94fdc9b7
[6] Dive into Anything, Reddit. [Online]. Available: https://www.redditinc.com/
[7] How to Read a 10-K/10-Q, SEC.gov, 2021. [Online]. Available: https://www.sec.gov/oiea/investor-alerts-and-bulletins/how-read-10-k10-q
[8] Xiao Ding et al., “Knowledge-Driven Event Embedding for Stock Prediction,” *Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics*, pp. 2133-2142, 2016. [Google Scholar] [Publisher Link]
[9] Amazon Neptune, Amazon Web Services. [Online]. Available: https://aws.amazon.com/neptune/
[10] JanusGraph. [Online]. Available: https://janusgraph.org/
[11] Neo4j Documentation, Neo4j. [Online]. Available: https://neo4j.com/docs/
[12] Getting Started, Apache Tinker Pop. [Online]. Available: https://tinkerpop.apache.org/docs/current/tutorials/getting-started/
[13] Graph Database Platform Overview, TigerGraph. [Online]. Available: https://www.tigergraph.com/product/
[14] Cypher Query Language, Neo4j. [Online]. Available: https://neo4j.com/developer/cypher/
[15] SPARQL, World Wide Web Consortium (W3C). [Online]. Available: https://www.w3.org/2001/sw/wiki/SPARQL
[16] Financial Modeling Prep, Datarade. [Online]. Available: https://datarade.ai/data-providers/financial-modeling-prep/profile
[17] Apple (AAPL) Stock Price, Quote, News and History, Nasdaq. [Online]. Available: https://www.nasdaq.com/market-activity/stocks/aapl
[18] NASDAQ Composite Index (COMP) Latest Quotes, Charts, Data and News, Nasdaq. [Online]. Available: https://www.nasdaq.com/market-activity/index/comp
[19] Getting Started with Amazon Neptune, Amazon Web Service. [Online]. Available: https://docs.aws.amazon.com/neptune/latest/userguide/graph-get-started.html
[20] Named Entity Recognition, CoreNLP. [Online]. Available: https://stanfordnlp.github.io/CoreNLP/ner.html
[21] Batched Graph Classification with DGL, Deep Graph Library, 2019. [Online]. Available: https://www.dgl.ai/blog/2019/01/25/batch.html
[22] PyG Documentation, Pytorch_Geometric Documentation. [Online]. Available: https://pytorch-geometric.readthedocs.io/en/latest/

[23] Pankaj Gupta, "Leveraging Machine Learning and Artificial Intelligence for Fraud Prevention," *SSRG International Journal of Computer Science and Engineering*, vol. 10, no. 5, pp. 47-52, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[24] Mayorga Lira Sergio Dennis, Laberiano Andrade-Arenas, and Miguel Angel Cano Lengua, "Credit Risk Analysis: Using Artificial Intelligence in a Web Application," *International Journal of Engineering Trends and Technology*, vol. 71, no. 1, pp. 305-316, 2023. [CrossRef] [Google Scholar] [Publisher Link]