# KSE30 Companies Network Analysis

CS-343: Graph Data Science
Habib University
By: Abdullah Junejo (07154) Alizain Merchant (07565)

Final Report
-
Date: May 7, 2024

## 1 Introduction

In this project, we explore the intricate and ever-evolving landscape of the Karachi Stock Exchange 30 (KSE30) Index, which represents the top 30 corporations in Pakistan by market capitalization and liquidity. Our goal is to employ graph database technology to dissect and elucidate the complex web of relationships and interdependencies among these premier entities. Utilizing graph databases enables us to represent and probe these relationships with unmatched intuitiveness and efficiency. Specifically, by mapping the KSE30 companies and their principal executives and board members as nodes within a graph database, we seek to shed light on the influence of these individuals within the market, explore their network connections, and analyze overall market trends. For this purpose, we will utilize Neo4j, gathering our data directly from the Pakistan Stock Exchange through the PSX Data Portal (https://dps.psx.com.pk/).

## 2 Data Collection

The Data was manually collected by going into the PSX Data Portal and then KSE30 where we saw different market performers. Upon clicking the company and noting down its information one by one, we were able to put together a .csv file for this (Attached). This task can be furthered on KSE100 as well after successful demonstration of this miniature project.

# 3 Data Modelling/Import

Here is our initial Graph Model that we will use. Hoever, his may very well be improvised upon in the future!

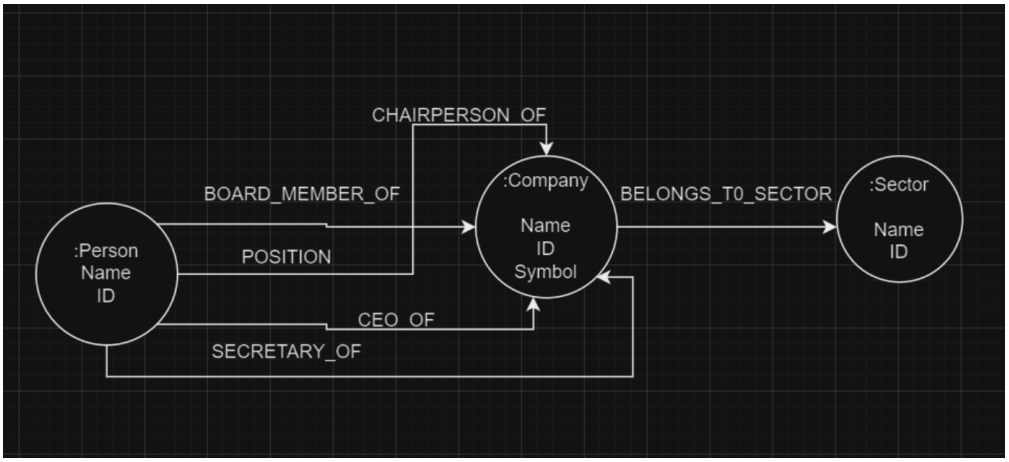


Figure 1: Initial Model

After Creating Project, Installing Graph Data Science/APOC Libraries, dropping the file in import folder, we worked on this script and used it to import data our csv file according to our afforementioned model.

```
1   LOAD CSV WITH HEADERS FROM 'file:///KSE_30_Formatted.csv
        ' AS row
2   WITH row
3   WHERE row.Board of Directors IS NOT NULL AND TRIM(row.
        Board of Directors) <> ""
4   UNWIND SPLIT(row.Board of Directors, ";") AS
        directorDetails
5   WITH row, TRIM(directorDetails) AS directorDetails
6   WHERE directorDetails <> ""
7   WITH row,
8        TRIM(SPLIT(directorDetails, "(")[0]) AS DirName,
9        TRIM(SPLIT(SPLIT(directorDetails, "(")[1], ")")[0])
             AS position
10  WHERE row.Year IS NOT NULL AND TRIM(row.Year) <> ""
11    AND row.Company Sector IS NOT NULL AND TRIM(row.
        Company Sector) <> ""
12    AND row.Secretary IS NOT NULL AND TRIM(row.Secretary)
        <> ""
```

```
13        AND row.Chairperson IS NOT NULL AND TRIM(row.
              Chairperson) <> ""
14        AND row.Company Symbol IS NOT NULL AND TRIM(row.
              Company Symbol) <> ""
15        AND row.Company Name IS NOT NULL AND TRIM(row.Company
              Name) <> ""
16        AND row.Company CEO IS NOT NULL AND TRIM(row.Company
              CEO) <> ""
17        AND position IS NOT NULL AND TRIM(position) <> ""
18
19      MERGE (company:Company {name: row.Company Name, symbol:
            row.Company Symbol})
20      MERGE (sector:Sector {name: row.Company Sector})
21      MERGE (chairperson:Person {name: row.Chairperson})
22      MERGE (secretary:Person {name: row.Secretary})
23      MERGE (ceo:Person {name: row.Company CEO})
24
25      WITH company, sector, chairperson, secretary, ceo,
            DirName, position
26      MERGE (director:Person {name: DirName})
27      MERGE (director)-[:BOARD_MEMBER_OF {position: position
            }]->(company)
28      MERGE (company)-[:CHAIRPERSON_OF]->(chairperson)
29      MERGE (company)-[:SECRETARY_OF]->(secretary)
30      MERGE (company)-[:CEO_OF]->(ceo)
31      MERGE (company)-[:BELONGS_TO_SECTOR]->(sector);
```

After this, it became aparent that there were multiple edges of "Board Member" from the same Person Node to the Same Company Node but having different edge property called position. So we decided to merge them so that future queries are easier and performance is not compromised. The property of position was also in a way merged. Different positions were seperated by a backslash.

```
1  //Step 1: Match and collect positions for each director-
       company pair
2  MATCH (director:Person)-[r:BOARD_MEMBER_OF]->(company:
       Company)
3  WITH director, company, COLLECT(DISTINCT r.position) AS
       positions, COLLECT(r) AS rels
4
5  //Step 2: Delete the existing relationships
6  FOREACH (rel in rels | DELETE rel)
7
8  //Step 3: Create a new single relationship with all
```

```
    positions combined
WITH director, company, REDUCE(s = "", position IN positions
    | s + (CASE WHEN s = "" THEN "" ELSE "\\" END) +
    position) AS combinedPositions
MERGE (director)-[newRel:BOARD_MEMBER_OF]->(company)
SET newRel.positions = combinedPositions
```

This essentially merged Multiple Board of Director edges into one if they are related to the same company.

# 4 Graph Statistics

This initial statistic provides a snapshot of the various sectors represented in the KSE30 index, the total number of companies, and detailed counts of key executive roles across all companies. This fundamental data lays the groundwork for deeper analyses, offering a clear view of the breadth and scope of leadership within these influential companies.

```
    //Statistics 1:
MATCH (sector:Sector)
WITH COLLECT(sector.name) AS Sector Names

MATCH (company:Company)
WITH Sector Names, COUNT(company) AS Total Number of
    Companies

MATCH (director:Person)-[:BOARD_MEMBER_OF]->(company)
WITH Sector Names, Total Number of Companies, COUNT(DISTINCT
    director) AS Total Number of Board Members

MATCH (chairperson:Person)<-[:CHAIRPERSON_OF]-(company)
WITH Sector Names, Total Number of Companies, Total Number
    of Board Members, COUNT(DISTINCT chairperson) AS Total
    Number of Chairpersons

MATCH (ceo:Person)<-[:CEO_OF]-(company)
WITH Sector Names, Total Number of Companies, Total Number
    of Board Members, Total Number of Chairpersons, COUNT(
    DISTINCT ceo) AS Total Number of CEOs

MATCH (secretary:Person)<-[:SECRETARY_OF]-(company)
RETURN Sector Names, Total Number of Companies, Total Number
    of Board Members, Total Number of Chairpersons, Total
```

```
    Number of CEOs, COUNT(DISTINCT secretary) AS Total Number
     of Secretaries
```

This second analysis reveals directors who hold positions across multiple companies (Highest 2 as of now), which could give a general idea about their influence across corporations and potential conflicts of interest. Next query is about tracking chairpersons who currently oversee multiple companies ($\exists$ 1 such chairperson as of now) further sheds light on the leadership dynamics. The third query tries to find the company that sits at the center of the most extensive connections, either through direct executive roles such as secretary, chairperson, CEO or through other types of affiliative relationships such as Board Member. Such a company typically acts as a central node within the network which we hope to unravel and confirm in our future explorations. Next we try to Highlight the sector that includes the most companies within the KSE30 offers insights into which part of the economy is most saturated. This information is crucial for steering investment strategies, creating economic policies (regarding subsidy/taxation), and conducting market analysis, as it reflects the sector's economic significance and its potential for expansion or risk.

```
1     //Statistics 2:
2     // Collect data for board members belonging to multiple
         companies
3     MATCH (director:Person)-[:BOARD_MEMBER_OF]->(company:
         Company)
4     WITH director, COUNT(DISTINCT company) AS Number of
         Companies
5     WHERE Number of Companies > 1
6     WITH COLLECT(director.name) AS Multi Company Board
         Members
7
8     // Collect data for chairpersons of multiple companies
9     MATCH (chairperson:Person)<-[:CHAIRPERSON_OF]-(company:
         Company)
10    WITH chairperson, COUNT(DISTINCT company) AS Number of
         Companies, Multi Company Board Members
11    WHERE Number of Companies > 1
12    WITH COLLECT(chairperson.name) AS Multi Company
         Chairpersons, Multi Company Board Members
13
14    // Identify the company with the most affiliations
15    MATCH (person:Person)-[r]->(company:Company)
16    WITH company, COUNT(DISTINCT person) AS Number of
         Affiliates, Multi Company Board Members, Multi
```

```
        Company Chairpersons
17      ORDER BY Number of Affiliates DESC
18      WITH COLLECT({companyName: company.name, numAffiliates:
            Number of Affiliates})[0] AS Company With Most
            Affiliates, Multi Company Board Members, Multi
            Company Chairpersons
19
20      // Sector with the most companies
21      MATCH (company:Company)-[:BELONGS_TO_SECTOR]->(sector:
            Sector)
22      WITH sector, COUNT(DISTINCT company) AS Number of
            Companies, Company With Most Affiliates, Multi
            Company Board Members, Multi Company Chairpersons
23      RETURN sector.name AS Sector with Most Companies, Number
            of Companies,
24             Company With Most Affiliates.companyName AS
                   Company with Most Affiliations,
25             Company With Most Affiliates.numAffiliates AS
                   Number of Affiliated Persons,
26             Multi Company Board Members,
27             Multi Company Chairpersons
28      ORDER BY Number of Companies DESC
29      LIMIT 1
```

These statistics, derived from targeted Cypher queries, do more than just quantify the interconnections within the KSE30. They also provide a deeper qualitative understanding of corporate governance and sectoral prominence, laying a robust foundation for further analysis such as predictive modeling and trend examination. This enriched insight is instrumental in comprehensively understanding the dynamics that shape Pakistan's economic forefront.

# 5 Creating Further Data

Our data was mostly homogenous clusters of companies and their executives. Only few Board Members were part of 2 companies and only 1 chairperson. This shows that for our future analytics, it could be difficult to find a person to person shortest path. This is why we introduce government as stakeholders. It is a natural and well founded assumption that companies belonging to a sector will be regulated by the relavant government agencies. The company executives will have the meetings with such ministries and would be able to connect with their head, The minister. This could provide us a path to other clusters that was otherwise isolated. The Data was manually collected from Wikipedia and verified from various other official

government web sources.

```
1     //Creation of Ministries
2     CREATE (energy:Ministry {name: "Ministry of Energy"}),(
         sbp:Ministry {name: "State Bank of Pakistan"}),(
         itTelecom:Ministry {name: "Ministry of Information
         Technology and Telecommunication"}),(scienceTech:
         Ministry {name: "Ministry of Science and Technology"
         }),(foodSecurity:Ministry {name: "Ministry of
         National Food Security and Research"}),
3     (industries:Ministry {name: "Ministry of Industries and
         Production"}),
4     (commerceTextile:Ministry {name: "Ministry of Commerce
         and Textile Industry"}),
5     (health:Ministry {name: "Ministry of National Health
         Services, Regulations and Coordination"});
6
7     //Link Ministries to Sectors
8     MATCH (ministry:Ministry), (sector:Sector)
9     WHERE (ministry.name = "Ministry of Energy" AND sector.
         name IN ["Refinery", "Power Generation & Distribution
         ", "Oil & Gas Marketing Companies", "Oil & Gas
         Exploration Companies"]) OR
10           (ministry.name = "Ministry of Information
                Technology and Telecommunication" AND sector.
                name = "Technology & Communication") OR
11           (ministry.name = "Ministry of Science and
                Technology" AND sector.name = "Chemical") OR
12           (ministry.name = "Ministry of National Food
                Security and Research" AND sector.name = "Food
                & Personal Care Products") OR
13           (ministry.name = "State Bank of Pakistan" AND
                sector.name = "Commercial Banks") OR
14           (ministry.name = "Ministry of Industries and
                Production" AND sector.name IN ["Cement", "
                Cable & Electrical Goods"]) OR
15           (ministry.name = "Ministry of Commerce and Textile
                 Industry" AND sector.name = "Textile Spinning"
                ) OR
16           (ministry.name = "Ministry of National Health
                Services, Regulations and Coordination" AND
                sector.name = "Pharmaceuticals")
17    MERGE (sector)-[:REGULATED_BY]->(ministry);
18
```

```
19      //Creation of Minister Nodes and Linking to Ministries
            with Positions
20
21      //Minister of Energy
22      CREATE (musadikMalik:Person {name: "Musadik Masood Malik
            ", position: "Minister"})
23      WITH musadikMalik
24      MATCH (ministryEnergy:Ministry {name: "Ministry of
            Energy"})
25      MERGE (musadikMalik)-[:HEADS]->(ministryEnergy)
26
27      //Minister of Information Technology and
            Telecommunication
28      CREATE (shazaFatima:Person {name: "Shaza Fatima Khawaja"
            , position: "Minister"})
29      WITH shazaFatima
30      MATCH (ministryIT:Ministry {name: "Ministry of
            Information Technology and Telecommunication"})
31      MERGE (shazaFatima)-[:HEADS]->(ministryIT)
32
33      //Minister of Science and Technology
34      CREATE (khalidSiddiqui:Person {name: "Khalid Maqbool
            Siddiqui", position: "Minister"})
35      WITH khalidSiddiqui
36      MATCH (ministryScience:Ministry {name: "Ministry of
            Science and Technology"})
37      MERGE (khalidSiddiqui)-[:HEADS]->(ministryScience)
38
39      //Minister of National Food Security and Research
40      CREATE (ranaTanveer:Person {name: "Rana Tanveer Hussain"
            , position: "Minister"})
41      WITH ranaTanveer
42      MATCH (ministryFoodSecurity:Ministry {name: "Ministry of
             National Food Security and Research"})
43      MERGE (ranaTanveer)-[:HEADS]->(ministryFoodSecurity)
44
45      //Minister of Industries and Production
46      CREATE (ranaTanveerDup:Person {name: "Rana Tanveer
            Hussain", position: "Minister"}) // Duplicate node
            for same person different role
47      WITH ranaTanveerDup
48      MATCH (ministryIndustries:Ministry {name: "Ministry of
            Industries and Production"})
```

```
49      MERGE (ranaTanveerDup)-[:HEADS]->(ministryIndustries)
50
51      //Minister of Commerce and Textile Industry
52      CREATE (jamKamal:Person {name: "Jam Kamal Khan",
           position: "Minister"})
53      WITH jamKamal
54      MATCH (ministryCommerce:Ministry {name: "Ministry of
           Commerce and Textile Industry"})
55      MERGE (jamKamal)-[:HEADS]->(ministryCommerce)
56
57      //Health Secretary overseeing Ministry of National
           Health Services, Regulations and Coordination
58      CREATE (iftikharShallwani:Person {name: "Iftikhar Ali
           Shallwani", position: "Health Secretary"})
59      WITH iftikharShallwani
60      MATCH (ministryHealth:Ministry {name: "Ministry of
           National Health Services, Regulations and
           Coordination"})
61      MERGE (iftikharShallwani)-[:HEADS]->(ministryHealth)
62
63      //Resolving Transport Sector Issue
64
65      MATCH(sector:Sector {name: "Transport"})
66      MERGE(aviationMinistry:Ministry {name: "Ministry of
           Aviation"})
67      MERGE(maritimeMinistry:Ministry {name: "Ministry of
           Maritime Affairs"})
68      MERGE(p1:Person{name: "Khawaja Muhammed Asif", position:
            "Minister"})
69      MERGE(p2:Person{name: "Qaiser Ahmed Sheikh", position: "
           Minister"})
70      // Establish potentially applicable regulatory
           relationships
71      MERGE (sector)-[:REGULATED_BY]-(aviationMinistry)-[:
           HEADS]-(p1)
72      MERGE (sector)-[:REGULATED_BY]-(maritimeMinistry)-[:
           HEADS]-(p2)
73      return *
```

Now this all long script that is pretty inefficient and has plenty of corrections made to it afterwards is all about creating ministries that regulate sectors. A ministry can regulate many sectors, and a Minister can head multiple ministries (Max 2 in our database). In the case of Ministry of Health which is vacant at

the meoment the position is secratary (which for the sake of argument lets assume also works with the same minister in order to make our database relavant). The only problem remains is the Sector: Commercial Banks which is regulated by The State Bank of Pakistan which is headed by Board of Directors. One such director is Finance Secratary who reports to Finance Minister heading the finance minstry. Now we can resolve this issue with the following querry:

```
    //Dealing with SBP
MATCH (sbp:Ministry {name: "State Bank of Pakistan"})

//Creating board members and Secretary, and their link to SBP
MERGE (jameelAhmad:Person {name: "Mr. Jameel Ahmad", position: "Chairperson Governor"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (aliCheema:Person {name: "Dr. Ali Cheema", position: "Non-Executive Director"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (akbarZaidi:Person {name: "Dr. Syed Akbar Zaidi", position: "Non-Executive Director"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (najafKhan:Person {name: "Mr. Najaf Yawar Khan", position: "Non-Executive Director"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (fawadAnwar:Person {name: "Mr. Fawad Anwar", position: "Non-Executive Director"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (zahidEbrahim:Person {name: "Mr. Zahid Ebrahim", position: "Non-Executive Director"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (mahfoozKhan:Person {name: "Mr. Mahfooz Ali Khan", position: "Non-Executive Director"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (aliLatif:Person {name: "Mr. Muhammad Ali Latif", position: "Non-Executive Director"})-[:BOARD_MEMBER_OF_SBP]-(sbp)
MERGE (imdadBosal:Person {name: "Mr. Imdad Ullah Bosal", position: "Secretary Finance"})-[:BOARD_MEMBER_OF_SBP]-(sbp)

//Creating Finance Minister, Ministry and its link.

MERGE (p2:Person{name: "Muhammad Aurangzeb"})
MERGE (financeMinistry:Ministry {name: "Ministry for Finance and Revenue"})
```

```
MERGE (p2)-[:HEADS]->(financeMinistry)
Return *

//Linking Finance Secratary with a unique edge of reports to
    Finance Minister
MATCH(p1:Person{name:"Mr. Imdad Ullah Bosal"})
MATCH(p2:Person{name: "Muhammad Aurangzeb"})
MERGE(p1)-[:REPORTS_TO]->(p2)
return *

//Fixing Property
MATCH (muhammadAurangzeb:Person {name: "Muhammad Aurangzeb"
   })
SET muhammadAurangzeb.position = "Minister"
```

Now we have a working State Bank Node and its further Heirarchy uptil the finance minister. Now we just need to connect it with its relavant sector.

```
    MATCH(s:Sector{name:"Commercial Banks"})
    Match(b:Ministry{name:"State Bank of Pakistan"})
    MERGE(s)-[:REGULATED_BY]-(b)
```

Now we have all the ministers who should clearly know each other, we can simply add them to a common node so that information can easily flow making ease for the company executives.

```
    //Ending the Heirarchial Structure
    MATCH (p:Person)-[r:HEADS]-()
    MERGE (d:Cabinet{name:"The Cabinet of Pakistan"})
    MERGE (p)-[:REPORTS_TO]-(d)
    return *
```

This creates a unique labelled node called cabinet which is kind of the point for minsters to share information.

# 6 Results

All in all, Our focus extends beyond mere financial metrics to explore the governance structures, regulatory frameworks, and inter-ministerial influences that shape the operational landscapes of these entities. we aim to provide new insights into corporate governance and strategic alignments within Pakistan's economic framework.


Figure 2: Resultling Heirarchial Structure in Neo4j

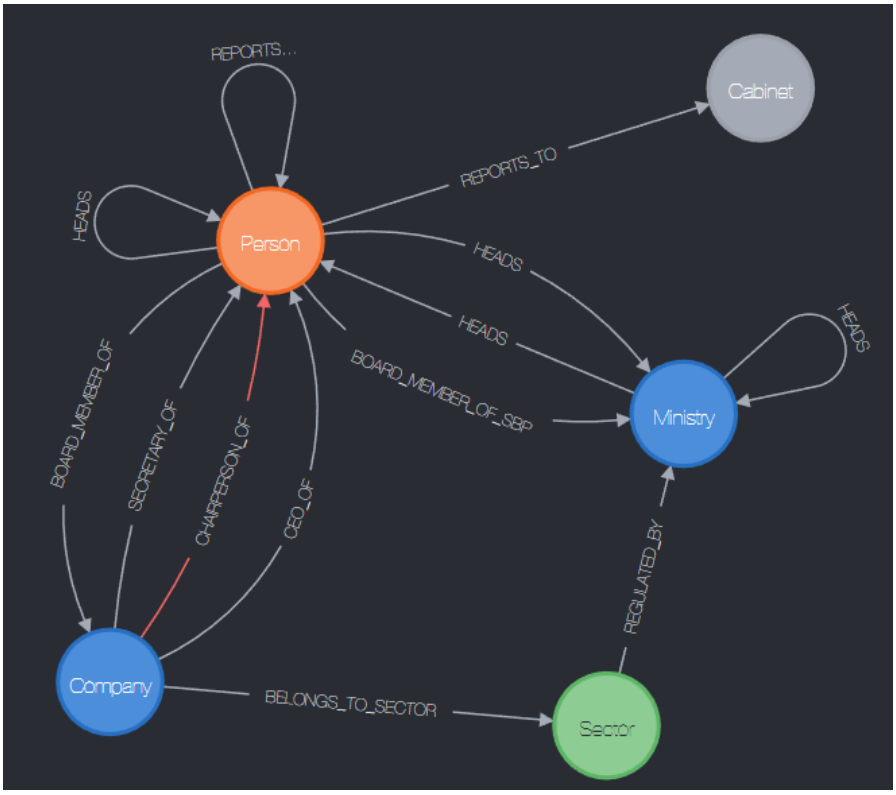


Figure 3: Final Model (CALL db.schema.visualization())

# 7 Graph Analytics

## 7.1 Shortest Path

```
    MATCH (p1:Person {name: 'Dr. Salman Faridi'})
    MATCH (p2:Person {name: 'Mr. Saad Amanullah Khan'})
    return shortestPath((p1)-[*]-(p2))
```

## 7.2 Centrality

For this we will have to create our projection first!

```
CALL gds.graph.project(
    'myGraph',
    ['Person', 'Company', 'Ministry', 'Sector', 'Cabinet'],
    ['BOARD_MEMBER_OF', 'HEADS', 'BELONGS_TO_SECTOR', '
       BOARD_MEMBER_OF_SBP', 'SECRETARY_OF', 'CHAIRPERSON_OF
       ', 'REGULATED_BY', 'REPORTS_TO', 'CEO_OF']
)
YIELD graphName, nodeCount, relationshipCount
RETURN graphName, nodeCount, relationshipCount
```

Next, we write querry for different types of centralities! That we will link to our Application:

```
    CALL gds.degree.stream('myGraph')
    YIELD nodeId, score
    RETURN gds.util.asNode(nodeId).name AS name, score AS
       degree
    ORDER BY score DESC
    limit 30
```

Here is the projection for Betweenness Centrality. Relationship Orientation is Important here: Betweenness centrality frequently necessitates taking into account the path in both directions. UNDIRECTED specifies that the algorithm considers paths regardless of their directional flow.

```
    CALL gds.graph.project(
        'boardMemberGraph',  // Name of the graph
        ['Person','Company', 'Ministry', 'Sector', 'Cabinet
           '],  // Node labels
        {
            BOARD_MEMBER_OF: {
                type: 'BOARD_MEMBER_OF',
```

```
            orientation: 'UNDIRECTED'
        },
        HEADS: {
            type: 'HEADS',
            orientation: 'UNDIRECTED'
        },
        SECRETARY_OF: {
            type: 'SECRETARY_OF',
            orientation: 'UNDIRECTED'
        },
        CHAIRPERSON_OF: {
            type: 'CHAIRPERSON_OF',
            orientation: 'UNDIRECTED'
        },
        REGULATED_BY: {
            type: 'REGULATED_BY',
            orientation: 'UNDIRECTED'
        },
        REPORTS_TO: {
            type: 'REPORTS_TO',
            orientation: 'UNDIRECTED'
        }
        // Include additional relationships as relevant
    }
)
YIELD graphName, nodeCount, relationshipCount
RETURN graphName, nodeCount, relationshipCount
```

```
CALL gds.betweenness.stream('boardMemberGraph')
YIELD nodeId, score
RETURN gds.util.asNode(nodeId).name AS name, score AS
   betweenness
ORDER BY score DESC
LIMIT 20
```

One surprsing thing that resulted is was that Sui Southern Gas Company had higher score than The Cabinet of Pakistan. This means that the company lies more often on paths. This implies that when people are trying to connect with other people, Sui Southern Gas Company plays a more of pivotal role than Cabinet of Pakistan. I believe this is because they are perhaps easier to approach compared to the cabinet yet have as much influence over the corporate world. Either that or we can assume that they just have huge PR which allows them to connect with multiple companies, organisations and have events for instance.

| name | betweenness |
|---|---|
| "Sui Southern Gas Company Limited" | 630.0 |
| "The Cabinet of Pakistan" | 443.0 |
| "K-Electric Limited" | 429.0 |
| "Dr. Imran Ullah Khan" | 364.0 |
| "Fauji Cement Company Limited" | 309.16666666666663 |
| "Pakistan Telecommunication Company Ltd" | 284.0 |

Figure 4: Betweenness Centrality Results

```
    CALL gds.graph.project(
    'closenessGraph',
    ['Person', 'Company'],
    {
        BOARD_MEMBER_OF: {
            type: 'BOARD_MEMBER_OF',
            orientation: 'UNDIRECTED'
        },
        CHAIRPERSON_OF: {
            type: 'CHAIRPERSON_OF',
            orientation: 'UNDIRECTED'
        },
        CEO_OF: {
            type: 'CEO_OF',
            orientation: 'UNDIRECTED'
        },
        SECRETARY_OF: {
            type: 'SECRETARY_OF',
            orientation: 'UNDIRECTED'
        }
    }
)
YIELD graphName, nodeCount, relationshipCount
RETURN graphName, nodeCount, relationshipCount
```

Now we look at Closeness Centrality, this time we have restricted it to Person and Company Nodes.

```
    CALL gds.closeness.stream('closenessGraph')
    YIELD nodeId, score
    RETURN gds.util.asNode(nodeId).name AS name, score AS
       closeness
    ORDER BY closeness DESC
    LIMIT 30
```

The results are fairly standard and almost all companies had score of 1.0. Person had lower than that.
Lastly, we look at PageRank Complexity:

```
    CALL gds.graph.project(
        'pageRankGraph',
        ['Person', 'Company', 'Ministry', 'Sector'],
        {
            BOARD_MEMBER_OF: {
                type: 'BOARD_MEMBER_OF',
                orientation: 'NATURAL'
            },
            HEADS: {
                type: 'HEADS',
                orientation: 'NATURAL'
            },
            REGULATED_BY: {
                type: 'REGULATED_BY',
                orientation: 'REVERSE'
            },
            CEO_OF: {
                type: 'CEO_OF',
                orientation: 'NATURAL'
            }
        }
    )
    YIELD graphName, nodeCount, relationshipCount
    RETURN graphName, nodeCount, relationshipCount
```

```
    CALL gds.pageRank.stream('pageRankGraph')
    YIELD nodeId, score
    RETURN gds.util.asNode(nodeId).name AS name, score AS
       pageRank
    ORDER BY score DESC
    LIMIT 20
```

(NEXT PAGE)

| name | pageRank |
|---|---|
| "Air Link Communication Limited" | 7.499114644577572 |
| "United Bank Limited" | 7.144354836534013 |
| "The Bank of Punjab" | 7.144354836534013 |
| "Mr. Muzzaffar Hayat Piracha" | 6.413879304616735 |
| "Mr. Muhammad Jawaid Iqbal" | 6.135492666113753 |
| "Mr. Zafar Masud" | 6.135492666113753 |

Figure 5: PageRank Results

We got different results this time, likely due to the High PageRank nodes being nearer to the important ones. This proves that PageRank considers the global context and does not rely solely on shortest path and its derivatives. However, this seems irrelavant in the context of this network. We can also think about this in this way that since they are near Ifluential nodes, they are likely to become influential in the future. For the currently influential nodes (Company/People), knowing which entities are viewed as influential by them and their peers (as highlighted by PageRank) versus being central by structure themselves (as shown by betweenness or closeness), they can do strategic planning accordingly.

# 8 Adding Features

```
1   CALL gds.graph.project(
2   'unifiedGraph',
3   ['Person', 'Company', 'Ministry', 'Sector', 'Cabinet'],
4   {
5       BOARD_MEMBER_OF: {type: 'BOARD_MEMBER_OF',
            orientation: 'UNDIRECTED'},
6       HEADS: {type: 'HEADS', orientation: 'UNDIRECTED'},
7       REGULATED_BY: {type: 'REGULATED_BY', orientation: '
            UNDIRECTED'},
8       CEO_OF: {type: 'CEO_OF', orientation: 'UNDIRECTED'},
9       REPORTS_TO: {type: 'REPORTS_TO', orientation: '
            UNDIRECTED'},
10      CHAIRPERSON_OF: {type: 'CHAIRPERSON_OF', orientation
            : 'UNDIRECTED'},
```

```
        SECRETARY_OF: {type: 'SECRETARY_OF', orientation: '
            UNDIRECTED'},
        BELONGS_TO_SECTOR: {type: 'BELONGS_TO_SECTOR',
            orientation: 'UNDIRECTED'}
    }
)
YIELD graphName, nodeCount, relationshipCount
RETURN graphName, nodeCount, relationshipCount
```

```
    CALL gds.pageRank.write('unifiedGraph', {
        writeProperty: 'pageRank'
    });
    CALL gds.betweenness.write('unifiedGraph', {
        writeProperty: 'betweenness'
    });
    CALL gds.closeness.write('unifiedGraph', {
        writeProperty: 'closeness'
    });
    CALL gds.degree.write('unifiedGraph', {
        writeProperty: 'degree'
    });
```

```
    CALL gds.beta.pipeline.linkPrediction.create('
        linkPredictionPipeline')
```

# 9 Tkinter Front End

Using Python's Library, Tkinter we designed a simple user-interface, where all of the above-mentioned section's querries are run with a click of a button. We used py2neo library to load our graph database (Password: 12345678).

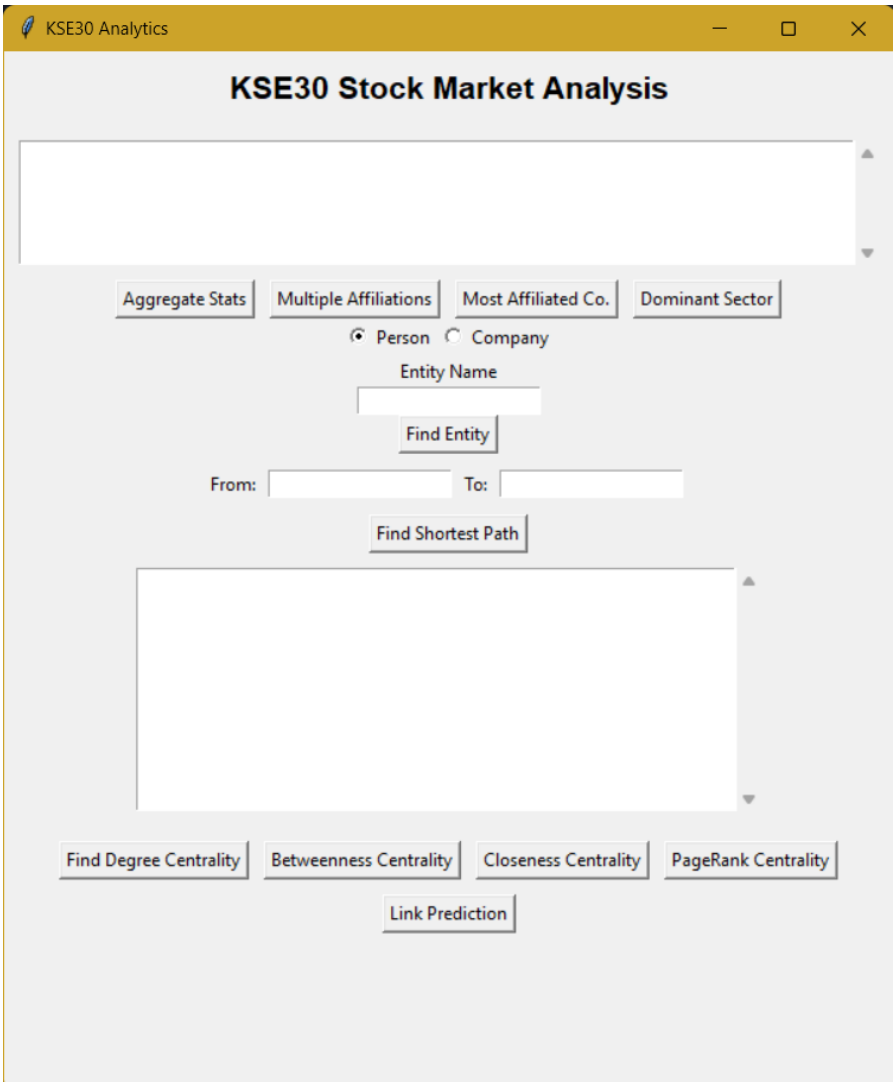


Figure 6: Tkinter-based Front End Design

# 10 References


1. Pakistan Stock Exchange. https://www.psx.com.pk/
2. Cabinet of Pakistan. https://en.wikipedia.org/wiki/Cabinet_of_Pakistan
3. Ministry of Foreign Affairs, Government of Pakistan. https://na.gov.pk/en/fmins_list.php