The Corporate-Political Nexus

A thesis submitted in partial fulfillment
of the requirements for

M.TECH MAJOR PROJECT
by

ABHISHEK AGARWAL

Entry No. 2014MCS2114

AMARTYA CHAUDHURI

Entry No. 2014MCS2117

Under the guidance of

Dr. AADITESHWAR SETH

Department of Computer Science and Engineering,
Indian Institute of Technology Delhi.

January 2016.

Certificate

This is to certify that the thesis titled **The Corporate-Political Nexus** being submitted by **ABHISHEK AGARWAL** & **AMARTYA CHAUDHURI** for the partial fulfilment of **M.Tech Major Project** in **Computer Science & Engineering** is a record of bona fide work carried out by them under my guidance and supervision at the **Department of Computer Science & Engineering**. The work presented in this thesis has not been submitted elsewhere either in part or full, for the award of any other degree or diploma.

Dr. AADITESHWAR SETH
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi

Abstract

In this project we try to facilitate the study of distribution of power across and within various Indian power institutions by analysing their inter-linkages, family trees, timelines, transactions, etc. to make more sense of the big political and corporate handshakes. In that direction, we envision to construct a system to collect data in this regard from various sources and integrate all of it into a single data store for others to use.

Our aim is to disseminate these findings to the mass society using a crowdsourced system, and use mass language for such a flow of information. and in this way, enhance governmental accountability and transparency using the notion of Open data and crowdsourcing.

Acknowledgments

We would like to express our heartiest gratitude to our supervisor Dr. Aaditeshwar Seth for guiding this work with utmost interest and scientific rigor. We thank him for setting high standards, giving us freedom to explore multiple facets of the problem and teaching us value of analytical thinking and hard work. We are also grateful to Manoj Kumar, Anirban Sen, and Dipanjan Chakraborty who have helped a great deal by providing technical guidance and support during difficult times and whose suggestions went a long way in making this work a reality. We would also like to thank our family and friends for their love and support.

ABHISHEK AGARWAL
AMARTYA CHAUDHURI

Contents

List of Figures

List of Tables

Chapter 1

Introduction

By common opinion, a democratic society in modern world is a misnomer - an illusion often given to the "ruled" party in a nation. Two centuries after Lincoln's definition of democracy, a government-for-the-people sadly is nothing but a collection of handful of population in control of basic resources of a country (natural or artificial). Leaders in military, politics, businesses, sports and arts thus become the actual voice of a country rather than the common mass. The distribution of power in these domains are often hereditary (among close ties in family) instead of elected representatives. Interactions among the persons of these important fields are also common for their effective functioning (or often to safeguard their self benefits.)

In this regard, accountability and transparency in government is one of the key requirements in order to obtain an ideal democratic society. Unfortunately the lack of proper knowledge about the politicians and corporates has led to new forms of "collective" dictatorship where public rights or voices are rendered ineffective.

To account for this growing problem of opacity we have tried to study the social network of Indian politicians and corporates and disemminate the information to the common public.

Problem Statement- Problem of forming the social networks of Indian Politicians and Corporates, visualizing and analysing them.

1.1 Objective

We intended to complete following tasks through our project-

- To **collect data from semantic web** (and other sources) to form a database (henceforth referred to as **"knowledge base"** / **"knowledge graph"**)

- To create **a neat, structured minimal error data collection** which otherwise is scattered at respective sources.
- To provide **a data mashup from different fields** to further help the academecians, journalists etc.
- To **monitor the top players in Indian society** - mainly in the spheres of politics and businesses in India.
- To **disemminate information to public** which brings about accountability and transparency.
- To seek answers to questions like -
  - *who were the big players in Indian politics and businesses?*
  - *Is there any influence (or possibility of it) of political field by a person in corporate field?*
  - *How important is one politician in a network of politicians (or a businessperson in a business network)?*
  - *Whom does actual power reside in a democracy?*

We believe that through our work, we will be able to show how such system of inter-disciplinary data helps to spread information and find patterns and discover more knowledge.

1.2 Motivation & Related Work

In his book **The Power Elite** [13], C. Wright Mills calls attention to the interwoven interests of the leaders of the military, corporate, and political elements of society and suggests that the ordinary citizen is a relatively powerless subject of manipulation by those entities. His book deals with the power elite in US. But the hierarchy he proposes is more or less the same across all countries. Power rests with the top one percent in an economy. We plan to create a watchdog for that one percent. One interesting list to

accompany this direction could be the Forbes list [3] of 147 companies that control everything.

French economist Thomas Piketty in his famous work **Capital in the Twenty-First Century** [14] focuses on wealth and income inequality in Europe and the United States since the beginning of the industrial revolution. He proposes a global system of progressive wealth taxes to help reduce inequality and avoid the vast majority of wealth coming under the control of a tiny minority. We plan to collect, integrate, visualize and open such data for Indian terrain to let data fanatics carry out sych works to understand this inequality.

British writer and historian Patrick French, in his book **India: A Potrait** [10] has stated many such interesting patterns in Indian politics where he argues that almost all of the young Indian politicians in the Indian Parliament are hereditary. Infact, patterns similar to this can be seen over the entire political Indian scene. One can find interesting overlaps, family ties, social links within these power houses. In a survey, **Who owns your media?** [7], we find that even the media is an entity of importance and most of the politicians tend to try pull their strings in this domain. Another interesting case could be Jayant Sinha's family tree [9] and their business holdings. He is the Minister of State for Finance and a Member of Indian Parliament and has links to lot of powerful companies.

Research along the area have been prominent across countries. **Sastry** [15] shows how crime and money play important role in Indian elections. In a related work **Vaishnav** [16] explain why do Indian parties elect criminal candidates and why they win. **Kapur** [11] connects the hidden relationships between politicians and builders. He argues that where elections are costly but accountability mechanisms are weak, politicians often turn to private firms for illicit election finance and that where firms are highly regulated, politicians can exchange policy discretion or regulatory forbearance for bribes and monetary transfers from firms

Works like what we propose have already been done for countries like USA, UK, Chile etc. We have examples like **LittleSis** [4], **Poderopedia** [8] where journalists, developers, analysts came together to put up profiles of important

entities, institutions of the society and highlighted the connections between them. Littlesis (opposite of Big brother) in one hand exists in USA from the political and economical data available there. Poderopedia is a similar site in Chile. These sites feature separate pages of people in power in USA, their connections to different institutions and other entities , work history, visualizations of the connections to educate masses etc. Other than producing awareness to people about the corporate- political connections, these sites also allow public to register and collaborate in data entry processes and has an API system to promote further use of their data for research purposes.

Such system in absence of digital data/ structured data and other human factors is difficult in India. But various local and national initiatives have been started. **Association for democratic Reforms** [1] for example has sites like **Myneta** [6] to disseminate information about political leaders of India.

Our vision is to produce a system similar in lines to the websites embedded with the power to query interesting connections, find interesting visualizations, and help raise suspicious issues.

The core two things that required our special attention when working with several data sources is the process of **modelling the database as a graph** and method of **resolving same entities from different data sources** . These two things are problems with extensive study of their own.

Entity resolution has been studied since 1946 by works of *Halber.L.Dunn* [citation]. Basic method is to match a pair of strings accurately to determine possible similar entities. Since then, several string matching algorithms has been used for this purpose. The **levenshtein algorithm** [citations] gives score based on no of edits to convert one string to another.It is escpecially useful for to deal with problem of misspelt records. Improving on that the **Jaro-Winkler** [citation] looks at matching characters within a small range while giving scores. This is suitable for short strings such as names and fits our purpose well. Another class of string matching algorithms looks at phonetics to resolve entities with similar sounding text. The **soundex algorithm** [citation] is one of the most well known. These algorithms use english language pronunciations to create an index of the text. For our

purpose, we have used the Jaro-Winkler and a variation of Soundex **(Double Metaphone)** [citations]

The graph model of the data is required since we are emphasizing the relationships among the data. Graph databases as a concept existed long since mid-1960s when **IBM IMS** [citation]supported graph structures in its hierarchical model. Graph databases allows one to give semantic queries over data relationships. A graph databases model the data as nodes and relationships among them as edges between nodes. Internally, they store the data as a relational table (**MariaDB** [citation]) or through document-value stores (**Neo4j, OrientDB** [citation]). For the purpose of our project we have used the Neo4j database to store the data.

1.3 Thesis Overview

The rest of the thesis has been divided into following -

1. **Social Network Creation.** -
2. **Design of Power Elites Web App.**
3. **Pattern Analysis**

Chapter 2

System Design

2.1 High Level Design

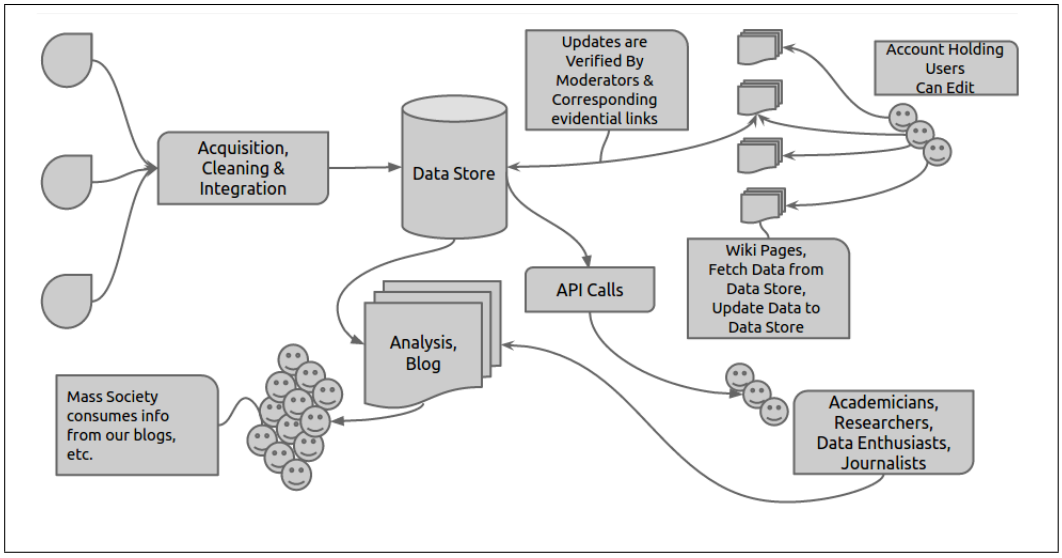

Figure 2.1: Big Picture

Above is the generic high level diagram of the entire system as we deduced from our study of the websites above. We have assumed three main types of users of here:

1. **General public** - Use the system for information and news. May also contribute towards data entry.
2. **Researchers & Academicians** - Use the site for using social network studies.
3. **Journalists (Media persons)** - Use the data to frame their news stories.

And keeping these in mind, the system accounts for following functions:

- **Data acquisition** - As getting structured data is often difficult system should have a mechanism to automate collection of data from various sources over internet.
- **Data Store** - There should be a central repository for whatever data collected. This repo will store the data in structured form. Care should be taken to keep its integrity, durability and non-redundancy.
- **Data Verification** - As the data is sensitive and important, provisions for verification for the input data has to be taken care of. This involves manual labor and system should incorporate this in the entire process.
- **Data Visualization** - A portal for the public display of data (in tables, visualizations etc.).
- **APIs** - To provide our data for use with other applications.

2.2 Technical design

Accordingly we have incorporated the functionalities in the following way:

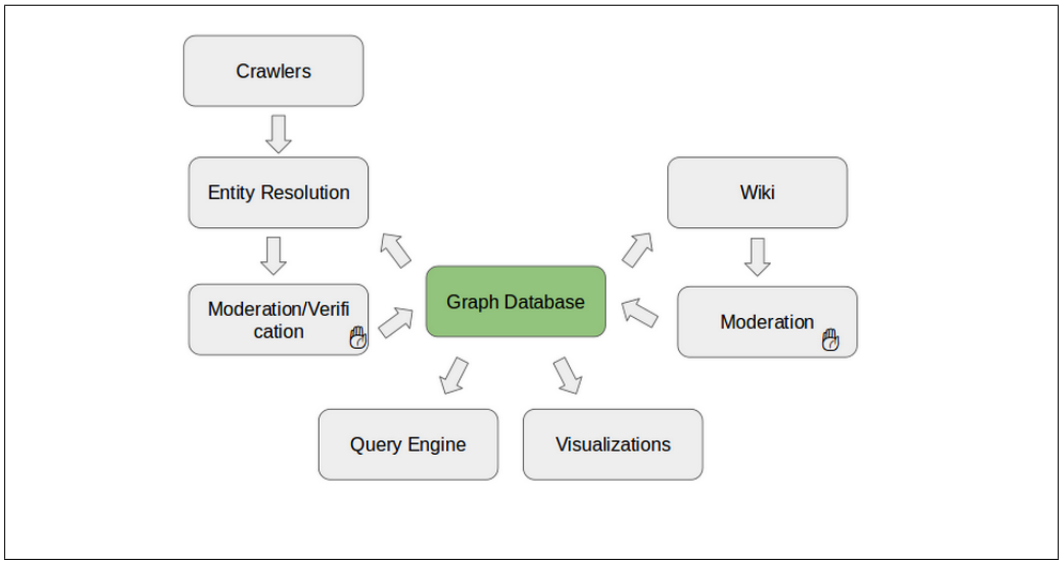

Figure 2.2: Proposed System

Components include:

- **Crawlers-Cleaner-Resolver** - As a part of the data acquisition module, the crawler crawls specific websites and scrapes data in format specified in a static file. Cleaner and resolver works on the scraped data. The cleaner makes the data more structured by keeping all data in specific format, discarding missing values etc. The resolver acts on the data to remove all duplicate entries (and merge similar entries) so to keep the data non redundant as possible.

- **Verifier**- The data coming from the crawler after resolution is verified by human moderator. Any to-be-updated information is first human moderated. Any new data is then fed to the graph database.

- **Graph database** - Acts as the data store for the system storing structured data with nodes as entities and edges as relationships.

- **Web portal(Query Engine, Visualization, Wiki)** - This is the final product that is directly visible to the users. All visualizations (graphical, tabular) are done here. It also has an wiki interface for an end user to add more entities/relations to the graph database. The portal also includes a query/search engine to show results as per specific user queries.

- **API** - Registered users can use this to read in data from the datastore in their application.

Chapter 3

System Pipeline

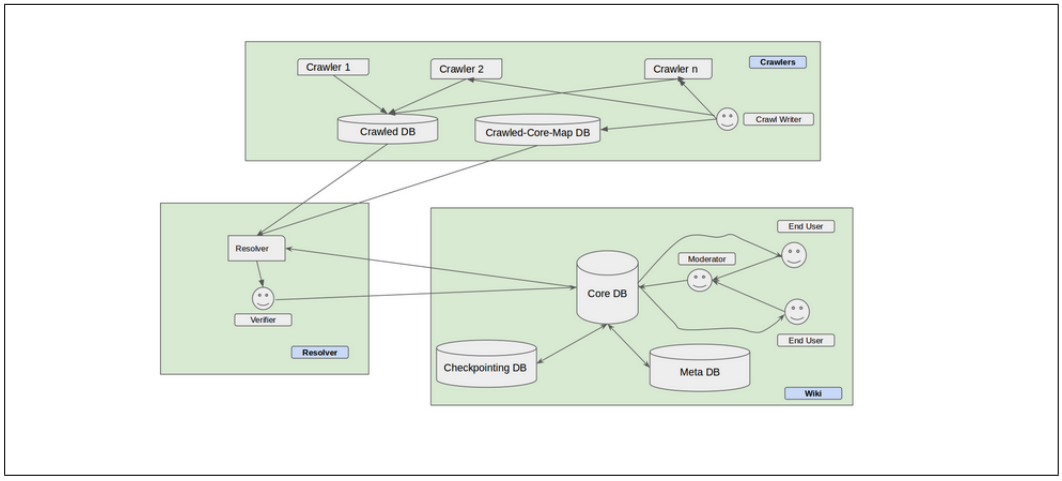

Figure 3.1: System pipeline

The total pipeline of the system can be seen in figure above. Overall, we have decomposed the system into 3 cohesive modules. In order of the data flow, they are -

Crawlers → Resolver → Wiki

3.1 Crawlers

This is the data acquisition module we have designed. Internally this module consists of the following components:

3.1.1 Crawler(s)

- These are basically python scrapy/beautiful soup/selenium scripts to crawl particular websites and srape data from them. The nature of the data (i.e - data types, labels) is mentioned in the ***Crawled-Core-Map*** database.

3.1.2 Crawled-Core-Map database

- Contains the type of the data, its name(or label) to be scraped from a website. This mapping information is used by crawlers to produce output data of specific type,label. This map along with the specific crawlers are written by the developers (***Crawl-Writers***)

3.1.3 Crawled database

- The raw output from the crawlers is written here. The schema of this database is largely dictated by the ***Crawled-Core-Map*** database. The data is then read from this database into the ***Resolver*** module.

3.1.4 Crawl Writers

- They are the human components of this module. These persons are developers who write specific scripts to scrape desired websites. They are also responsible for maintaining the data schema in ***Crawled-Core-Map*** database.

3.2 Resolver

This module works to fine tune the data scraped by the crawlers. All data collected thus far are mostly unformatted with duplicate or missing values. The function of this module is to process such data into a form suitable for the database. Its constituents are:

3.2.1 Resolver

- It takes the data from the Crawled database as input. The resolver can be further divided into two parts -

- **Cleaner** - The cleaner does the text normalization before the actual process for resolving begins. This includes out-of-format data, capitalization, missing-value issues which if not dealt with may cause the resolver to give low accuracy.

- **Resolver/Duplicator** - Function of resolver is two-fold. Firstly, it checks any duplicate records in the incoming data (i.e. - data from the crawled database) and removes if any. Then, it resolves the entries from the crawled data with that of the core data. And outputs all possible matchings to the verifier for verification/tagging.

3.2.2 Verifier

The authenticity of the data should be checked before it can be inserted into the ***Core Database***. The function of the verifier module and thus the human moderator is to tag the records outputted by the resolver module. These tagged records are the ones which are finally added to the ***Core Database***.

3.3 Wiki

This module is the web portal that is directly accessible to the end users. It consists of the interfaces that allow user to view, add, delete entities and their relationships interactively.

3.3.1 Core Database

This is the main data store of the system. Care has been taken to ensure that whatever data goes in is redundant, free of noise and authenticated. **Neo4j GraphDB** [12] is used in the backend for this.

- **Why Graph database preferred here?**
  Lot of brainstorming went in deciding to use Neo4j graph database. This is because a graph database stores the relations of records in the physical layer (unlike relational database) which makes faster retrieval

of connections of entities without joins. And most of the analysis in social networks involves reading in the relationships/connections between entities. Hence query results can be produced faster here without any complicated joins.

3.3.2 Meta Database

This contains the description of the data(format, source, type) being inserted in the database. This is especially useful when the data is authenticated against real-world information, so that every ounce of data in the core database is accounted for.

3.3.3 Checkpointing Database

Without a checkpoint, a Wiki is un-achievable. With every new update, a log of all changes is stored in this database. This is later used to roll back to a previous state to undo any new updates that occured.

3.3.4 Web Server

Background server that hosts the web application currently has 3 major functions.

- **Wiki + Visualization** - The basic functionality of the web application is to provide the users with profiles and connections of organizations and personas which they can edit. The server also maintains a mechanism for checkpointing any data received.
- **Query Engine** - To support the queries required for analysis of the network the server implements a query engine which takes queries of specific pattern and return results in tabular or visual format. This is achievable by Cypher query language which facilitates inquiring graph-like queries over Neo4j. These queries would go on the lines like: How are two entities connected? What is the shortest path between two entities? How far an influence of an entity goes over the graph?

Name	Age	Sex

Table 3.1: Sample data formats

Entity no.	Graph label	Graph props	Mysql props	Graph resolve props	Mysql resolve props
1	person	name,age,sex	name,age,sex	name	name

Table 3.2: Sample record of Crawled-Core-map database

- **Read API** - External applications can give GET requests to read data in json format. This aids research and analysis by end users.

3.3.5 Others

Human components in the system - *Users* and *Moderators*. Users include the wiki-users who edit the Wiki to enter any new/updated info they have. They have to provide an evidential link for any information they commit.
The job of moderators is to verify records entered by users in the Wiki. They can be experts in their domain, they need to cross-verify from trusted sources.

3.4 Example

Here we describe how the Verifier and Resolver actually work with a working example.

3.4.1 Verifier + Resolver

Let us say that the crawler crawls data and saves it in a format like in Table - 3.1 Also the crawler has an accompanying map-data information which looks like - 3.2
graph-props column have one to one order wise mapping with *mysql-props* column.
graph-resolve-props column have one to one order wise mapping with *mysql-resolve-props* column.

So, basically in the crawled data, each row represents a node with label :person and attributes (name, age, sex) in the core database.

1. Now, when a verifier logs-in a row from the 3.1 is fetched, then 3.2 is used to frame a query to search an entity with the name like in the row.
2. Matching nodes are suggested after the query to the verifier.
3. If verifier selects one of the propsed nodes, the resolved node is updated.
4. Else if the verifier doesn't find any matching node, a new node correspoding to the selected row is created and inserted in the core database.

3.4.2 Wiki + Visualizations

Profiles

Following is a typical profile in the Wiki-

The figure above shows the Wiki page for industrialist Naveen Jindal containing information about him. It also contains interfaces for any registered user to edit as happens in a Wiki. It also allows the user to show connections of the person.

Connections (Visualizations)

The figure above shows the connections for Minister of State for Finance Jayant Sinha. His connections include corporates firms like the **Aditya Birla Group** and **Pacific Paradigm Advisors** .

Other popular influence networks that our system shows is that of *Gandhi Family and their Corporate linkups*, *Jaydev Galla with his company with a large asset value*, *Kamal Nath with Moser Baer*, *Ravi Shankar Prasad with News 24 channel*.

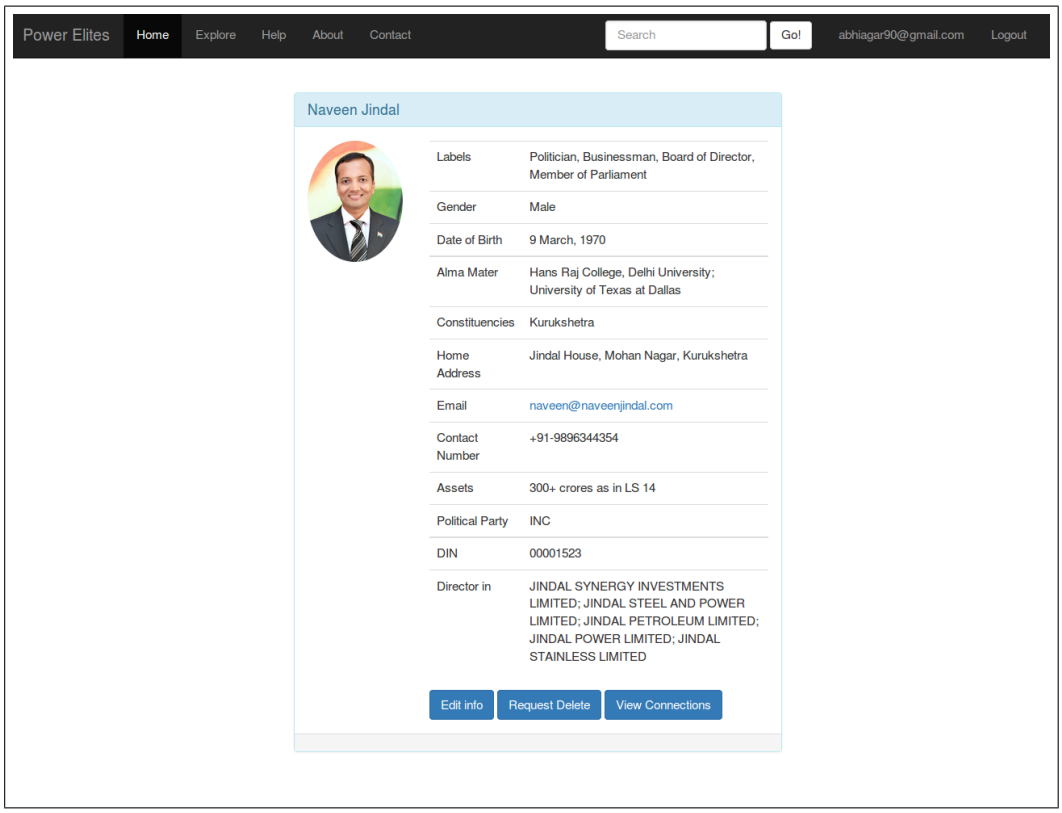

Figure 3.2: Naveen Jindal Profile

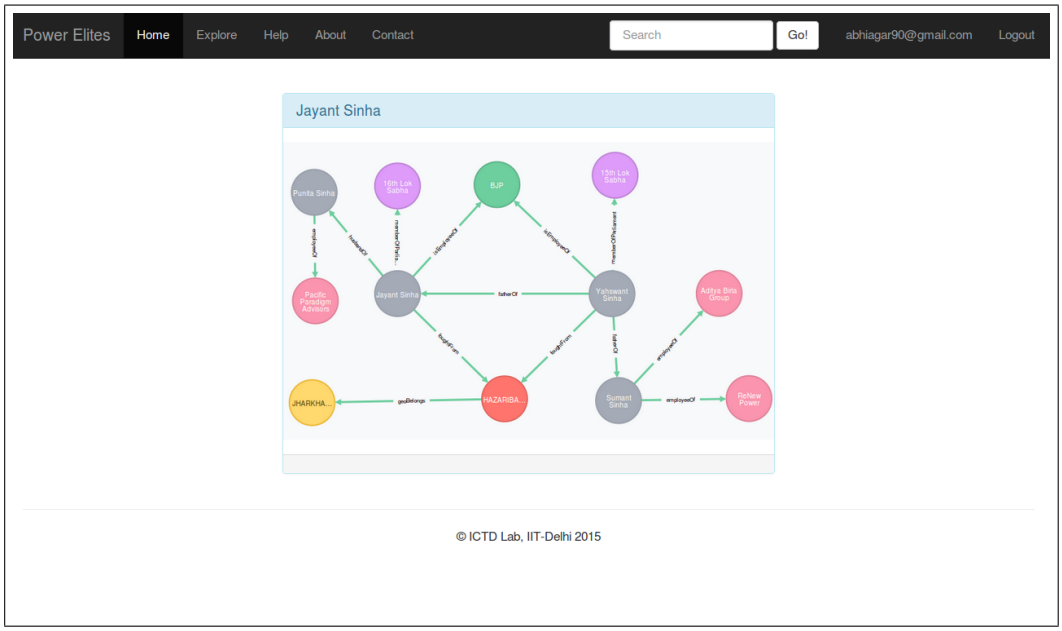

Figure 3.3: Jayant Sinha Connections

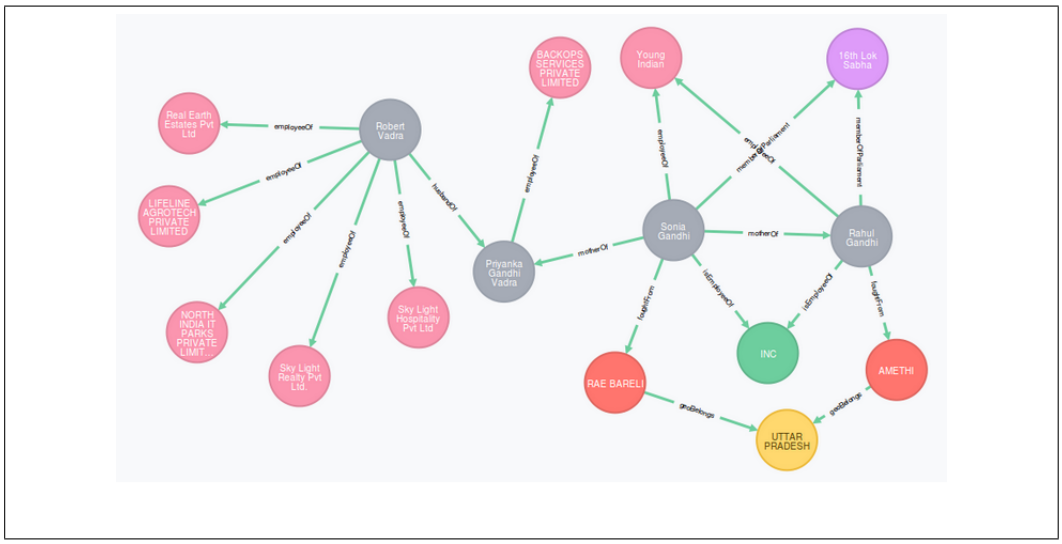

Figure 3.4: Gandhi family

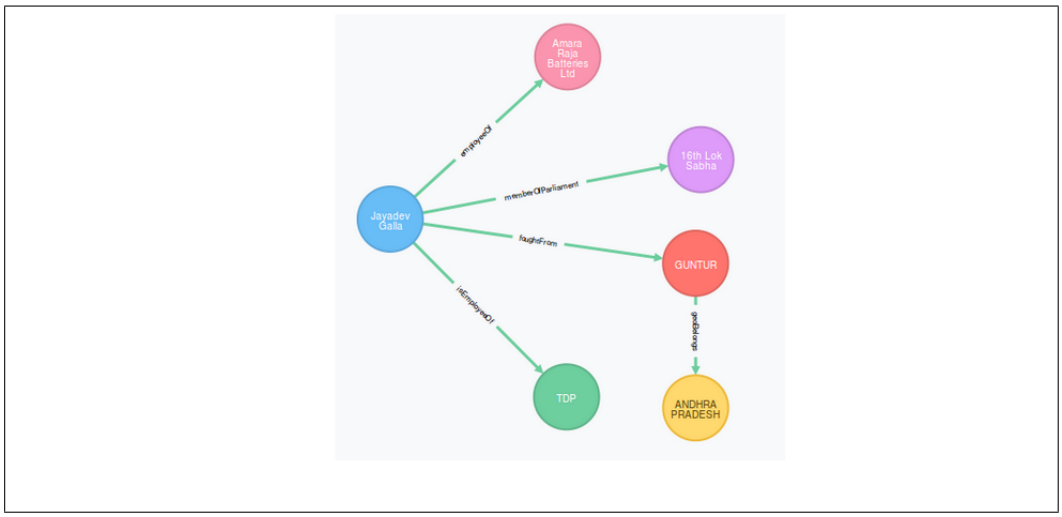

Figure 3.5: Jaydev Galla Connections

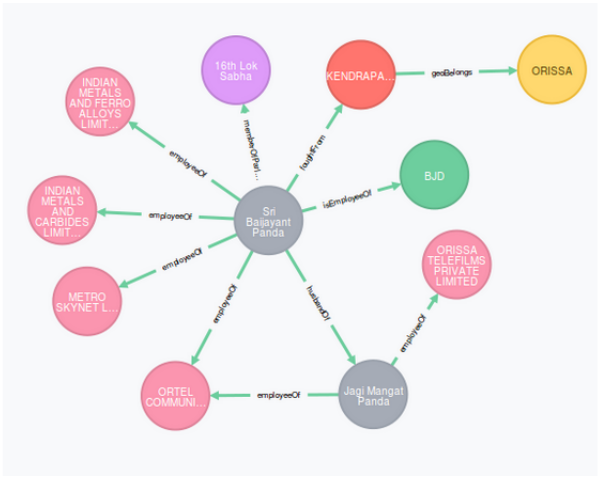

Figure 3.6: Jay Panda Connections

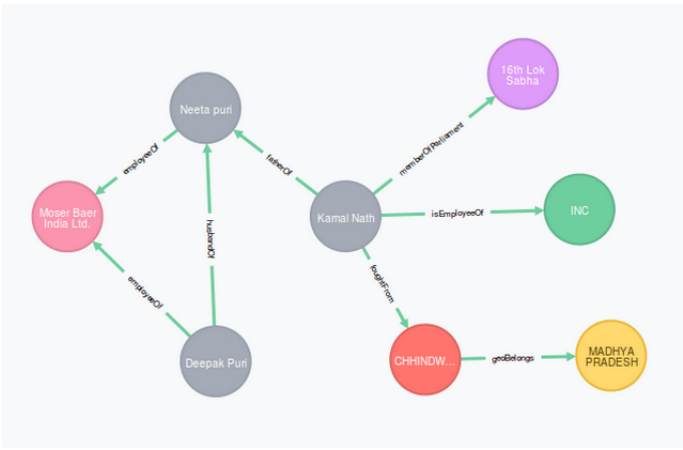

Figure 3.7: Kamal Nath Connections

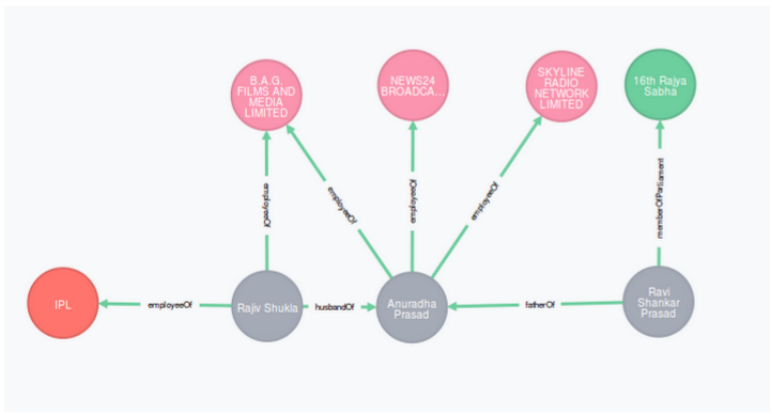

Figure 3.8: Ravi Shankar Connections

Chapter 4

Challenges

4.1 Datasets acquisition

Biggest challenge in mining political, bureaucratic, corporate data is the difficulty in getting relevant sources and structured information.

For our initial use cases we began by using political data from **MyNeta** [6]. We plan to enrich this data from LokSabha and Rajya Sabha archives. For corporate data, we used publicly available information from **CompanyWiki** [2] and **Ministry of Corporate Affairs** [5]. As the system progresses we plan to include retired IAS/IPS officers, PPP data, family data of politicians, donations in our growing dataset. We already have about 1000+ politicians, 60000+ business people, 90000+ companies in our current core DB.

4.2 Data Modelling

After getting some data the challenge was to model the data for the graph database. This would ensure listing of all possible entities and relationships of the system. Care was taken to form such relationships so that in the long run, queries supported by the system take optimum time to produce results. The prominent entites(nodes) and relationships(links) modelled so far:

Nodes/Entities-
(**Person** (uuid, name, address-location, DOB))
(**Politician** (uuid, name, mynetaid-electionname-year, constituency))
(**Alias** (uuid,name,context))
(**Party**(uuid,name,type:*"national,state,regional"*,start-year,HQ))
(**Company**(uuid,cin,name,location,income,expenditure,profit,value))

(**IAS** (uuid,name,DOJ,year,IAS-ID,posting-location))
(**Employee**(uuid,din,name,designation))
(**Govt-Body**(uuid,name,location))

Relationships/Links-
(**Politician**)→(**member-of**(id,type,years)→(**Party**)
(**Politician**)→(**member-of**(id,type,years)→(**Govt-Body**)
(**IAS**)→(**member-of**(id,type,years)→(**Govt-Body**)
(**Employee**)→employee-of(id,designation,years)→(**Company**)
(**Company**)→donated(id,cin,party,amt,year)→(**Party**)
(**Person**)→family(id,is-biological→(**Person**)
(**Person**)→profession(id,type)→(**Politician**)
(**Person**)→profession(id,type)→(**Corporate**)
(**Person**)→profession(id,type)→(**IAS**)
(**Person**)→aka(id)→(**Alias**)

4.3 Latency and Optimizations

Although implementation of basic system is ready, its performance suffers when single GET request to the Web server calls for multiple read requests to the core database. We are planning to alleviate this problem by indexing the database to optimize search time, forking multiple threads.

Chapter 5

Epilogue

5.1 Future Work

The constant aim would be to push as much data as possible and stablize the system as soon as possible. The query engine is very basic as of now. The further plan is to give users a platform to give structured queries to the system. We also plan to extend our sources of data, use information extraction algorithms to extract relationships from blogs, newspaper articles where the data is totally unstructured. The UI/accessibility of the system needs to be improved so that the verifiers, moderators, end-users can access the related pages/info readliy.

5.2 Conclusion

The system should be able to answer such questions as we move along:

- Preference between corporate donations to political parties.
- Preference between donation source locations and political parties.
- Ex-bureaucrats in corporate jobs and their political ties.
- Identification of shell companies/fraud companies in donation transactions and investments.
- Family trees of political candidates, corporate giants.

We are still at a very nascent stage. Trying to learn in our work and hopeful to create something beneficial for the society in future.

Chapter 6

Diagrams

Bibliography

[1] Association of democratic reforms - non-governmental organization for electoral and political reforms. http://www.adrindia.org/. 2015.

[2] Companywiki - find information about indian companies. http://www.companywiki.in/. 2015.

[3] Forbes - 147 companies that control everything. http://bit.ly/1O4BIgZ. 2015.

[4] Littlesis - profiling the powers that be. http://littlesis.org/. 2015.

[5] Ministry of corporate affairs - gateway to all services, guidance, and other corporate affairs related information in india. http://www.mca.gov.in/. 2015.

[6] Myneta - criminal and financial open data on politcians. http://myneta.info/. 2015.

[7] Newslaundry - who owns your media? http://bit.ly/1eY5Bj2. 2015.

[8] Poderopedia - collaborative platform that helps understand the relationships among important people, companies and organizations for chile. http://www.poderopedia.org/. 2015.

[9] Sinha family tree. http://bit.ly/1O4BIgZ. 2015.

[10] Patrick French. *India: A portrait*. Vintage, 2011.

[11] Devesh Kapur and Milan Vaishnav. Quid pro quo: Builders, politicians, and election finance in india. *Center for Global Development Working Paper*, (276), 2011.

[12] Justin J Miller. Graph database applications and concepts with neo4j. In *Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA March 23rd-24th*, 2013.

[13] C Wright Mills. *The power elite*. Oxford University Press, 1999.

[14] Thomas Piketty. Capital in the 21st century. *Cambridge: Harvard Uni*, 2014.

[15] Trilochan Sastry. Towards decriminalisation of elections and politics. *Economic and Political Weekly*, 4, 2014.

[16] Milan Vaishnav. *The Merits of Money and Muscle: Essays on Criminality, Elections and Democracy in India*. PhD thesis, Columbia University, 2012.