Searching Personnel Relationship from Census Data using Graph Database

Kay Thi Yar, Khin Mar Lar Tun
University of Computer Studies, Yangon
kaythiyar.ucsy@gmail.com

Abstract

Searching relationship among persons is more and more growing importance to acquire related information, genealogical relationship and personnel history. In this paper, the framework with three portions is proposed for exploring peoples' relationship from their personnel information. The first portion focuses on storage structure to store data in Graph database by representing persons as nodes and their attributes as properties. There are no predefined relationships or edges between person nodes. The second part focuses on searching relationships among persons. Graph database searching algorithm called Personnel Relationship Searching Algorithm is proposed. The last portion proposes Deductive Reasoning Algorithm to define two persons' relationship based on search results of match domain (or) attributes. For example, if the match domain is relatives, the relation may be consanguine, such asGrandFather,Eldest_Son,Youngest_Son_In_Law, Khame_Khamet, etc. If the match domain is organization (or) hobby (or) religion, the relation may work in same organization, etc.

1. Introduction

In today's world, the great majority of people around the global are citizens of the information society. Every nation has vast amount of census data and analysis of these data is the value for nation as source citations, correlating and corroborating sources, relevance or findings contradictions. These census data may relate in any form based on family group records, friendship, co-worker and etc. For contacting and exchanging information among people, searching relationship between persons is necessary and vital matter. In this paper, our nation, Myanmar's census data and including facts of NRC card are used as source citations for searching relationship. These data involve person name, date of birth, gender, occupation, parent names, relationship with householder, NRC number and detail parent's family records including jobs and etc. NRC number is the unique identification number for every citizen in Myanmar. The aim of this paper is to explore graph database structure that can support effective storage structure for peoples' connected information, to study efficient searching algorithm that can find the relationship from separated person nodes and to provide deductive reasoning for defining the indirect relationship among persons. Although many researchers are interested in searching relationship in social network [6], the research that searches the relationship between persons in real world is rarely found. Most of the relationships searching researches have been developed based on the predefined relationship among every node. Therefore, this proposed framework is wanted to develop for searching relationship based on the personnel information of separated nodes stored in graph database. The main purpose of finding relation between persons is to trace and discover the criminal cases (searching personnel relationship between A and B for the case like missing MH370 flight), to explore the properties of certain person for corruption case and to exchange information among people. The rest of this paper is organized as follows: section 2 expresses some relevant works; section 3 discusses the proposed system; section 4 describes the graph storage structure; section 5 provides graph searching algorithm; section 6 presents deductive reasoning algorithm. Finally, conclusion and future work are stated in section 7.

2. Related work

Several researches have been developing for searching relationship between persons and making recommendation mostly in social network like Facebook. J. Rivero et.al.[1] proposed the SoS-ACO (Sense of Smell - Ant Colony Optimization) algorithm to search relationship between social network members by improving the classical ACO algorithm [2]. L. Adamic and E. Adar [3] also search a network of friend based on Clubs Nexus of Stanford University and find short paths in students' social network using local information about their immediate contacts such as HP Labs email network, to pass the message from the sender to receiver with shortest path. S. Nuanmeesri

et.al. [4], search genealogical information in family tree proposing the Parent Bidirectional Breadth Algorithm (PBBA) and identify relationship name by applying rules based system with English rules. The authors [4] focus to solve some problems such as moving of daughters to their husbands' house because of marriage and to know consanguine relationships of two people who do not live in the same house. K. Keller et.al.[5], presented the Family Tree Visualization using the conical-shaped family tree layout to properly display all the necessary and useful information. As a family tree gets bigger, visualizing it becomes a more difficult task. However, no searching is provided in their paper. These researchers proposed searching connections among persons in social network and family tree by using graph traversal algorithms through the predefined relationships between nodes in graph database. In this paper, person nodes are separately created with no predefined relationships among them where relationships may be direct links, links with one or more intermediate persons and disconnected persons. Therefore, Personnel Relationship Searching Algorithm and Personnel Relationship Deduction algorithm are applied to find relationships between those complex relations among persons.

3. Proposed System

The proposed system's framework is shown in figure 1. The census data of Myanmar is used. The proposed framework include three parts; graph storage structure, graph searching and reasoning. For storage structure, the personnel information is stored as graph structure with persons as nodes by using Neo4j graph database. For graph searching, the user needs to provide two persons' names to search their relation using Personnel Relationship Searching Algorithm. The results of the search algorithm include relationship types such as father, daughter, brother, wife, etc… or personnel information. For reasoning, Personnel Relationship Deduction Algorithm is used to define the final relation for given two persons using the deductive reasoning rules. The final output of the proposed system can be one of the following; family members (e.g; father, son, sister, …) or Relatives (e.g; son in law, khamekhamet, …) or general relationship (e.g; same group of work organization, nationality, religion, …).

(i) Graph Storage Structure

Neo4j graph database is utilized to store personnel information. It can deliver well performance when handling highly interconnected data compared to traditional relational database [8]. Graph database usually consists of nodes and relationships. It is used to store people's information such as Name,Father_Name,Mother_Name,Husband/Wife_Name,Elder_Sister_Name,Middle_Son_Name, NRC_No,Occupation,Address,Phone_No,Hobby, etc. However, relationships between person nodes are not predefined in this database. Predefining every relationship among person nodes such as Father, Mother, Eldest_Daughter, Husband, Wife, etc is time consuming and impossible in practical. Therefore, consanguine relationship or association between persons will be searched through their persons' attributes data.

(ii) Graph Searching Algorithm

The personnel relationship searching algorithm is utilized to find relationship between two persons. This searching focuses on matching every domain (or) attributes of each node for given two persons' names. For example, the searching algorithm starts matching one of the given values with every property (attributes such as name, occupation, address, father name, mother name, sibling name, etc.) of each node. The search results may be list of nodes that match in one of those properties. Then the second given value is searched again in the data nodes by matching with every properties. The list of nodes that match with the second given values are obtained as search results. Among two result lists of nodes, the common node and the intermediate nodes are searched to find match relationship types for given two persons. If the common node and the intermediate nodes are not found, personnel information of two persons is retrieved to define the general relationship among them.

(iii) Deductive Reasoning Algorithm

After searching the common node or intermediate nodes that are related to given two persons, the relationship is defined by using the deductive reasoning algorithm. For example, if the matched domain is relative or consanguine, types of relatives will be defined. If the matched domain is on general attributes like work or hobby, the co-worker or friend relation will be defined.

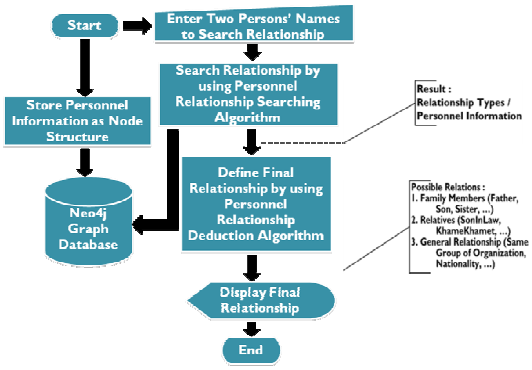

Figure 1. Proposed system overview

4. Graph Storage Structure

4.1. Graph Database

A graph database is the storage of graph-oriented data structures with nodes, edges, and properties to represent and store data. It is an occurrence based and schema less. Store data and relationships as they are encountered [9]. It can store complex and dynamic relationships of highly connected data like personnel information. Storage is optimized for the traversal of the graph, without using an index when following edges. Fast deep traversal instead of slow SQL queries that span many tables joins. Graph databases are used in many application domains like Social Networking and Recommendations, Calculating Routes, Network and Cloud Management, Master Data Management, Geospatial, Bioinformatics, Content Management, Security and Access Control [8]. Besides, there are many kinds of graph database such as Neo4j, OrientedDB, Titan, DEX, FlockDB, InfiniteGraph, HyperGraphDB, GraphBase, and InfoGrid. Among them, Neo4j property graph database is used for this system with the following reasons: Neo4j builds upon the property graph model. Both nodes and relationships can hold any desired properties called key-value pairs. It has no rigid schema, node-labels and relationship-types can be defined arbitrary by users. It is reliable and fast for managing and querying highly connected data. It is a powerful traversal framework for high-speed graph queries [12].

4.2. Neo4j Graph Database Storage Structure

Neo4j graph database is a browser based command driven client, like a web-based shell environment. Graph data can be retrieved by using Cypher query language which retrieves the data based on the predefined relations among nodes. Query results are displayed with tabular form and graph visualization form containing nodes and relationships [12]. Using Cypher query language, person nodes and properties of each person are created as below:

Create (n1 : PersonnalInfo {Name : "U Zay Yar", Father_Name : "U Thein Aung", Mother_Name : "Daw Khin Kyi", Wife_Name : "Daw Khin Kyawe Kyi", Eldest_Son_Name : "Mg Zar Ni Linn", ……. , NRC_No : "5/SaKaNa(N)072217", Race : "Burma", Occupation: "Township Education Officer", Department: "Administration", Organization : "Education Office, Sagaing", …….. , Hobby : "Playing Football", Favourite_Song : "Be Successful",Favourite_Movie:"PeeMak", Alive_Death : "Alive" }),

In this proposed system, personnel information of each person is created as node structure as shown in figure 2. Nodes contain properties called key-value pairs such as Name: "U Zay Yar", Father_Name: "U Thein Aung", etc. No predefined relationships are created according to the connections between person nodes. In this system, three depths of ascendants and descendants are defined for persons who are parents of the age between 50 and 70.

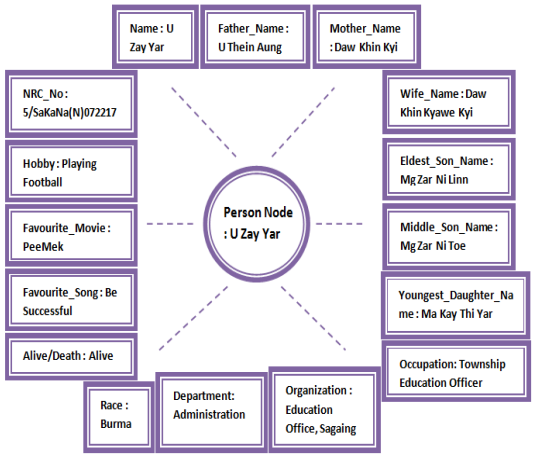

Figure 2. Creating a person node and some of its' properties (attributes)

5. Graph Searching Algorithm

Personnel Relationship Searching Algorithm performs the relationship searching process by dividing into four cases. Text mining and information extraction concepts are applied in retrieving related nodes that have same properties with given two search persons. From those same properties nodes, named entity recognition concept is used to retrieve the property keys or relationship types between those nodes. The output of this search algorithm may be relationship types or same properties between given two persons.

Case 1 searches direct relationship, e.g., family members such as Mother and Son (or) Father and Son.

In such case, no deductive reasoning is necessary to define the relation.

Case 2 searches indirect relationship between two persons with common person or intermediate persons. For example, Father and Daughter_In_ Law, KhameKhamet have indirect relationship from father and son, son and wife, where the common person is son and based on the common person's information, the relationship among them is constructed. For intermediate person case, there are one or more common persons for give two persons. For this case, intermediate persons are retrieved based on the properties of the related persons of First Person and Second Person. The relationship between intermediate persons for First Person is firstly defined using the deductive reasoning algorithm. Then the relationships between First Person, Second Person and the intermediate persons are defined again using the deductive reasoning algorithm.

Case 3 searches relationship between two persons who are not relatives. If the First Person and Second Person are not genealogically related, the relationships among these persons are retrieved based on same properties such as same occupation, same organization, and same hobby and so on.

Case 4 defines for general relationship between given two persons. For any given two persons, they can be general relationship such as group of religion or nationality or race or marital status. There may not be defined relationship such as genealogical or same properties relation.

The Personnel Relationship Searching Algorithm is shown in Algorithm 1.

6. Deductive Reasoning Algorithm

Reasoning is the process of forming conclusions and judgments from facts or premises. It is the ability to coherently think from perceived premise to a logical conclusion [10]. There are many types of reasoning; most common reasoning techniques are Deductive Reasoning, Inductive Reasoning, Reductive Reasoning and Fallacious Reasoning. In this paper, deductive reasoning is used to define the relation between persons based on their relationship types, properties values and reasoning rules. Deductive reasoning, also called logical deduction is the process of reasoning from one or more general statements (premises) to reach a logically certain conclusion [7]. It originates from the philosophy, mathematics and is the most obvious form of reasoning. It works from the more general to the more specific. Sometimes this is informally called a top-down approach. The deductive reasoning begins with a general hypothesis or known fact and creates a specific conclusion from that generalization. If all premises are true, the terms are clear, and the rules of deductive logic are followed, then the conclusion is necessarily true [11].

Personnel Relationship Searching Algorithm

Input : Two Persons' Names, Personnel Information Graph Database D
Output : Two Persons' Final Relationship, Relationship Types Rt, Same Personnel Information P_Info, Same General Personnel Information P_Info

```
Begin
  For each person node in D
    Case 1 : searching direct relationship between two persons
    who are family members
        If (each property of First Person FP equals with Name
        of Second Person SP){
          Direct = 1;
          Return Final Relationship Between FP and SP
         }
    Case 2 : searching indirect relationship between two
    persons with common person or intermediate persons
        If (direct != 1) {
          1. Retrieve Person Nodes that include FP.
          2. Insert Person Nodes to ArrayList1 AL1.
          3. Retrieve Person Nodes that include SP.
          4. Insert Person Nodes to ArrayList2 AL2.
          5. Find CommonPerson CP between AL1 and AL2.
          6. If (CommonPerson CP != Null){
               Retrieve Properties of CP.
               Retrieve PropertyKey between FP and CP.
               Retrieve PropertyKey between CP and SP.
             }
          7. If (CommonPerson CP == Null){
               Retrieve Properties of persons in AL1
                that match with Person Names in AL2.
                Retrieve Intermediate Persons IP from AL1 and
                AL2.
                Retrieve PropertyKeys between FP, SP and IP.
             }
          8. Reasoning (RelationshipTypes Rt);
          }
    Case 3 : searching relationship between two persons who
    are not relatives
        Retrieve Same Properties between FP and SP.
        Reasoning (P_Info);
    Case 4 : searching  general relationship between FP and SP
        Retrieve Same General Properties between FP and SP.
        Reasoning (P_Info);
  EndFor
End
```

Rt =>Relationship Types
P_Info => Same Personnel Informaion
D => Personnel Information Graph Database
FP => First Person
CP => Common Person
SP => Second Person

Algorithm 1. Personnel Relationship Searching Algorithm

Personnel Relationship Deduction Algorithm

Input : RelationshipTypes Rt, Same Personnel Information P_Info, Same General Personnel Information P_Info, Rule Database D
Output: Specific Conclusion C

For each rule r in D
  if r □ Rt C then
    /*(if Relationship Types Rt match with rule r then give conclusion of that rule)*/
      return C;
  else if r □ P_Info C then
    /*(if Same or General Personnel Information P_Info match with rule r then give conclusion of that rule)*/
      return C;
  endif
EndFor

Rt =>Relationship Types
P_Info => Same Personnel Informaion
D => Rule Database
r => rule
C => Specific Conclusion

Algorithm 2. Personnel Relationship Deduction Algorithm

In this relationship search, this paper proposes deductive reasoning algorithm, shown in algorithm 2 which takes the output of search algorithm 1 as input that includes relationship types such as youngest sister, wife, father or same personnel information such as occupation, organization, hobby or same general personnel information such as nationality and religion. All possible deductive reasoning rules are predefined for genealogical relationships such as father, mother, siblings, grandfather, and khame khamet and general relation such as work in same organization or same religion. For example:

a) Deductive reasoning rule for genealogical relationships like grandfather is defined as follows:

- "if Person1 is father of Person2" and "if Person2 is father of Person3" then "Person1 is grandfather of Person3"

b) Deductive reasoning rule for same properties (attributes) such as same organization and occupation,

- "if organization and/or occupation of Person1 are same with organization and/or occupation of Person2," then "Person1 and Person2 relates that they work in same organization and/or same occupation"

c) Deductive reasoning rule for same general properties such as same nationality and religion,

- "if nationality and/or religion of Person1 are same with nationality and/or religion of Person2," then "Person1 and Person2 are in same group of nationality and/or religion"

Thus, the output of Personnel Relationship Deduction Algorithm may be genealogical relationships between any given two persons such as great grandson, father in law, twin sister, etc. or same or general personnel properties such as same organization or religion.

7. Case Study for Searching Two Person's Relationship

Searching two person's relationship from set of data nodes in graph database is discussed with three case studies. Those case studies are tested on real world personnel data set with 3000 data nodes and each node has more than 25 properties. Neo4j Graph database, Cyber query that retrieves the nodes' properties and two proposed algorithms; searching and reasoning are also utilized to experiment these test cases. Some experiments have done on searching relationship in the Census data which includes the facts about Myanmar citizenship such as NRC no, personal profiles and other additional information such as hobby, favorite music and movies. This proposed framework is implemented using Java and Neo4j Community Edition.

Case Study 1: Searching Indirect Relationship from the Common Person.

In this case study 1, two persons: Ma Kay Thi Yar and Ma Khin Me Me Ohm who have Eldest Sister In Law relationship is searched by using the algorithms as follows:

- The first process of searching algorithm retrieves the set of data nodes that have similar names with search persons' name. Then, the common person nodes are selected from those result nodes.
- When the common person is found, its properties are retrieved to define the relation between search persons. In figure 3, two related person nodes are listed for each search person and then common person and relationship types are retrieved.
- The results of searching algorithm are relationship types: "Youngest Sister" and "wife". Then, deductive reasoning algorithm is applied to define the relationship between two search persons.
  - The deductive reasoning rules; "if Person1 is Youngest Sister of Person2" and "Person3 is Wife of Person2" then "Person3 is Eldest Sister In Law of Person1".

Therefore, final result: "Eldest Sister In Law" is defined for Ma Kay Thi Yar and Ma Khin Me Me Ohm.

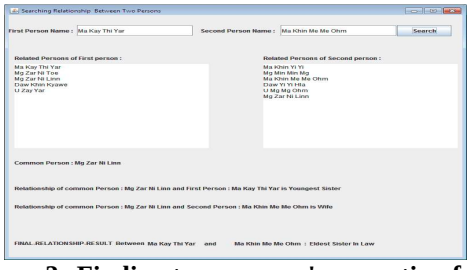

Figure 3. Finding two persons' connection from common person

Case Study 2: Searching Indirect Relationship from Intermediate Persons.

In this case study 2, two persons: U Zay Yar and U Mg Mg Ohm who have Khame Khamet relationship is searched by using the algorithms as follows:

- The first process retrieves the data nodes that have similar names with search persons. Then search for common person node in both lists of result nodes. When no common person node is found, intermediate person nodes and their properties are needed to find. In figure 4, related person nodes are listed for each search person. The search algorithm outputs the relationship types: "Wife" and "Father".
- When the intermediate person nodes and the relationship types are found, deductive reasoning algorithm is applied to define the relationship between two search persons. The deductive reasoning rules;
  - "if Person1 is Father of Person3" and "if Person4 is Wife of Person3" and "if Person2 is Father of Person4" then "Person1 and Person2 are khame khamet" .
- Therefore, final result: "Khame Khamet" Relationship is defined between U Zay Yar and U Mg Mg Ohm.

Case Study 3: Searching Relationship from Same Properties of Two Persons who are not Relatives.

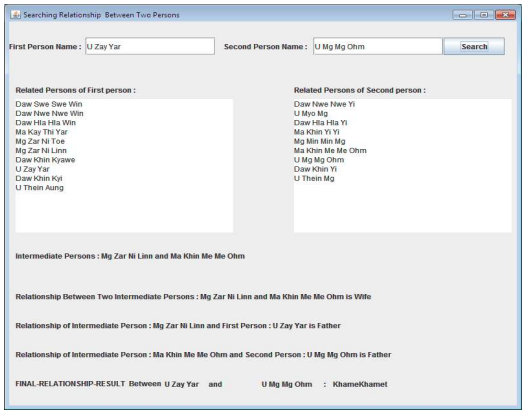

Figure 4. Finding two persons' connection from intermediate persons

In this case study 3, two persons: Dr. Khin Mar Lar Tun and Ma Kay Thi Yar who have same properties such as same department and/or organization are searched by using the algorithms as follows:

- The first process retrieves the data nodes that have similar names with search persons. There is no common person or intermediate person nodes in those result nodes. Therefore, the properties of two search persons are retrieved to define the relation between them. In figure 5, two related person nodes are listed for each search person.
- The searching algorithm outputs the same properties of department and organization. The deduction algorithm defines the relation as work in same organization and department.

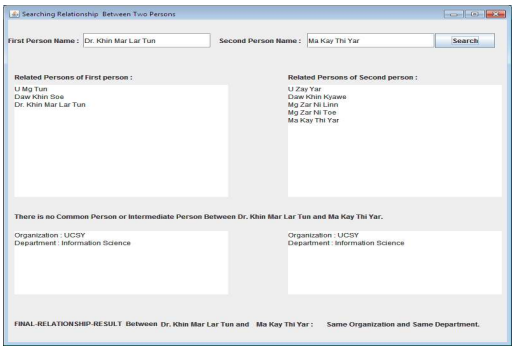

Figure 5. Finding two persons' connection from same properties

According to the above test cases, the proposed algorithms can precisely define relationship for any given two persons among graph structure of data set. By giving just two persons' names to be searched, the relationship among them is searched without needing any additional personnel information. The relationship is defined based on the predefined reasoning rules for family members (e.g., genealogical relationship like great grandmother, Sister in Law, Grand Nephew and so on) and general relationship (e.g., work in same organization, group of same religion and nationality). However, relationships are not predefined manually between person nodes when nodes are created in graph database. In this system, the person names are only considered for search keywords. Sometimes the input person names may be same or different. For same name case, there may be two situations; two persons may refer to single person otherwise they may be different persons. Moreover, the given two persons may have cyclic relationship and/or have same common or intermediate persons. For these cases, the proposed algorithms can search and define the accurate relationship for given two persons based on only their names. This relationship searching process is useful to explore relationship between long-lost persons, family members and relatives, to trace the personnel history, to relate and exchange information with each other that have same group of field and interest. However, the time complexity of algorithms is not yet considered and the algorithms' performance and/or efficiency are not yet defined in this framework.

8. Conclusion and Future Work

To cap it all, this paper proposes the framework for searching personnel relationship whatever they are connected via one or more intermediate persons or

same information persons. With the support of graph database, complex and dynamic relationships of highly connected data like personnel information can easily be created. Personnel Relationship Searching Algorithm also makes easy to search on large graph data with person nodes without predefining the relationships among persons. Due to the effectiveness of personnel relationship deduction algorithm, any relationship among persons can simply and accurately be inferred. For further development, since current implementation is done on the platform of homogeneous database such as single graph database, experiments over the heterogeneous databases like relational, Nosql and graph database will be implemented to compare the query processing time and to find optimal running time. Furthermore, the correctness and performance of proposed algorithms will be verified and proved with Hidden Markov Model or Conditional Markov Model.

References

[1] J. Rivero et. al. , "A bio-inspired algorithm for searching relationships in social networks", 2011 International Conference on Computational Aspects of Social Networks (CASoN), IEEE, 978-1-4577-1133-6/11, 2011.

[2] J. Rivero, D. Cuadra, J. Calle and P. Isasi, "Using the ACO algorithm for path searches in social networks", Applied Intelligence, DOI: 10.1007/s10489-011-0304-1, 2011.

[3] L. Adamic and E. Adar, “How to search a social network”, Social Networks vol. 27 (3), pp. 187-203, 2005.

[4] S. Nuanmeesri et. al., “Genealogical information search by using parent bidirectional breadth algorithm and rule based relationship”, in (IJCSIS) International Journal of Computer Science and Information Security, Vol. 6, No. 3, 2009.

[5] K. Keller et. al., “Family tree visualization”, 2011.

[6] M. Buerli, 2012. “The current state of graph databases”, Department of Computer Science, Cal Poly San Luis Obispo, mbuerli@calpoly.edu.

[7] M. Alberti, "Deductive Reasoning Agents", March, 19, 2012.

[8] I. Robinson, et al. 2013. “Graph Databases”, June 2013: First Edition.

[9] http://en.wikipedia.org/wiki/Graph_database

[10] http://en.wikipedia.org/wiki/Reasoning

[11] http://en.wikipedia.org/wiki/DeductiveReasoning

[12] http://www.neo4j.org