

Predictive Analysis of Personnel Relationship in Graph Database

Kay Thi Yar

Information Science Department
University of Computer Studies, Yangon
Yangon, Myanmar

Khin Mar Lar Tun

Information Science Department
University of Computer Studies, Yangon
Yangon, Myanmar

Abstract— Nowadays, social networks are so popular and widely used in all over the world. In addition, searching personal information of each person and searching connection between them (peoples' relation in real world) becomes interesting issue in our society. In this paper, we propose a framework with three portions for exploring peoples' relations from their connected information. The first portion focuses on the Graph database structure to store the connected data of peoples' information. The second one proposes the graph database searching algorithm, the Modified-SoS-ACO (Sense of Smell - Ant Colony Optimization). The last portion proposes the Deductive Reasoning Algorithm to define two persons' relationship. This study reveals the proper storage structure for connected information, graph searching algorithm and deductive reasoning algorithm to predict and analyze the personnel relationship from peoples' relation in their connected information.

Keywords— *personnel information, graph storage structure, graph searching algorithm, deductive reasoning algorithm*

I. INTRODUCTION

Every country collects a vast amount of data about their nations 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. In this paper, our nation, Myanmar's census data is used as source citations for searching relationship between two distinct persons based on unique National Registration Card (NRC) Number. The Myanmar census 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 algorithms that can find the relationship from graph database and to provide deductive reasoning for searching the undirected relationship among persons. In today's world, the great majority of people around the global are citizens of the information society. For contacting and exchanging information among people, searching relationship between persons is necessary and vital matter. To do that, graph databases are utilized to store connected data like personnel information as graph structure with persons as nodes and relationships between them as edges

by using Neo4j graph database. They can deliver well performance when handling highly interconnected data compared to traditional relational databases [8]. Relational databases are schema based, define the structure in advance

and deal poorly with connected data. They are generally efficient unless the data contains many relationships requiring joins of large tables. Due to the necessity of relationship nature, graph database technology is emerged for these types of queries, and providing in millisecond responses. It is the best technology for dealing with complex, semi-structured, densely connected data and substantially quicker than relational and nosql data store. Graphs are widely used to model social networks and the use of graphs to solve real world problems is becoming necessary [6]. However, when the size of the graphs is bigger and bigger, the time to search paths in large graphs should be fast. To do that, some types of preprocessing are needed to change the structure of the graph and to reduce the number of nodes/links over which the search has to be done. After that, they apply classical algorithm such as A-Star, Dijkstra, etc. with some modifications to handle the new structure of the graph. However, these preprocessing can take a large amount of time for the huge graphs. Thus, considering the topology and the adaptability characteristic, there is an algorithm that could be highly recommended: Ant Colony Optimization Algorithm [1]. It has been used in a lot of applications like job scheduling in grid computing, creating personalized guides within museums using mobile devices, determining medical diagnoses, travelling sales man problem, etc.). Therefore, searching algorithm related to graph database like Modified-SoS-ACO (Sense of Smell - Ant Colony Optimization) algorithm is applied to find relationship among persons within the few number of processing steps and shortest time. Based on this Algorithm result, the final personnel relationship will be deduced by using Personnel Relationship Deduction Algorithm.

The rest of this paper is organized as follows: Section 2 describes the subject or matters motivate me to do this system. In section 3, some relevant works are expressed. A brief discussion of the system is presented and the system overview is also shown in section 4. Section 5 describes the graph storage structure and graph searching algorithm is explained in section 6. Section 7 also presents deductive reasoning algorithm. Finally, conclusion and future work are stated in section 8.

II. MOTIVATION

Although many researchers are interested in searching relationship in social network like Facebook, the research that searches the relationship between persons in real world is rarely found. Every country needs basic information on its residents for purposes of planning, development and improvement of their quality of life. Census data and including facts for creating NRC card is the basic information for our country, Myanmar. By using these data, it is necessary

and essential to find related information between persons for the following reasons. The main purpose 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. Thus, graph database storage structure becomes one of the options to store data that are connected in some way. Searching or finding connection between data from the graph database, we need graph related searching algorithms. In order to make decision and to define the accurate relationship among those data, the concept of deductive reasoning is necessary to define. These points have led to motivate us to do this work.

III. RELATED WORK

Several researches have been developing for searching relationship between persons and making recommendation in social networking. A new algorithm based on nature SoS-ACO (Sense of Smell - Ant Colony Optimization) is proposed to search relationship between social network members in A Bio-Inspired Algorithm for Searching Relationships in Social Networks [1]. This algorithm improves the classical ACO algorithm when it is applied in huge graphs. It is able to search paths in the graphs of Social Networks with a huge number of nodes and edges, with a high clustering degree, and with a low number of steps to reach the destination node from a start node. In using the ACO algorithm for path searches in social networks [2], this algorithm is proposed in detail. In ACO algorithm, two concepts are used such as the defining the food node as meeting point for ants and the diffusion of the food odor is to decrease the number of nodes that find the food node. Using above two concepts, this algorithm can make easy to reach the path from the start node to the end node. Moreover, it also makes easy to search on large graphs without the need to change the structure of the graph or carry out a complex pre-processing. To find short paths in students' social networking website using local information about their immediate contacts, two methods are discussed in how to search a social network [3]. For searching in HP Labs email network, three possible strategies are used for passing the message from the sender to receiver with shortest path. They are best connected, closest to the target in the organizational hierarchy and located in closest physical proximity to the target. For searching in a network of friend based on Clubs Nexus of Stanford University, the possible strategies are both undergraduate, both graduate, or one of each; same year, a year apart, or more than a year apart; both male, both female, or one of each; same or different residences; same or different major/department and school. For searching genealogical information in family tree, Parent Bidirectional Breadth Algorithm (PBBA) is used to find consanguine relationship between two persons and to identify relationship name by applying rules based system with English rules [4]. There are some problem occurrences such as moving of daughters to their husbands' house occurs again and again because of marriage and to know consanguine relationships of two people who have not live in the same house. To solve these problems, PBBA is developed by combining the Bidirectional

approach and Breadth First Search Algorithm.

Parent direction is the only important direction for searching. Thus, this algorithm is faster than the original Bidirectional approach and Breadth First Search. Family Tree Visualization presents how to effectively display the layout of the temporal data in family tree [5]. Many family trees fail to properly display all the necessary and useful information. As a family tree gets bigger, visualizing it becomes a more difficult task. Therefore, conical-shaped family tree layout is applied. In their layout, the ancestors and descendants are laid out around a centered person. Circular axis highlights the age differences between certain nodes and is centered on the root node. This allows the users to see which nodes are how far in age from the root node and also supports dynamic interaction with the family tree. These related papers propose searching connections among persons in social network and family tree by using various ways. In this paper, we explore the graph database storage structure, graph searching algorithm and deductive reasoning to find personnel relationship in real world. Graph database storage structure is used to support effective storage structure for complex and interconnected personnel information. When the size of the graphs is bigger and bigger with the growth of the population, the time to search paths of graphs should be fast. Moreover, relationships between persons may be direct links, no direct links with one or more middle persons or disconnected persons. Therefore, Modified-SoS-ACO (Sense of Smell - Ant Colony Optimization) algorithm and Personnel Relationship Deduction algorithm is necessary to applied for searching various kinds of relationships between those complex persons. Besides, we proposes framework as shown in figure 1 and discovers practical results in further work.

IV. PROPOSED SYSTEM

Our proposed framework as shown in figure 1 bases on the census data, including facts for creating NRC card, personal profile and other additional information such as hobby, favorite music and movie. This framework is intended to realize with three portions such as graph storage structure, graph searching and reasoning. These personnel information are stored as graph structure with persons as nodes and relationships between them as edges by using Neo4j graph database. Modified-SoS-ACO searching algorithm is utilized to find whether relationship between two persons or not. For Related Graph, if relationship exists among them, the final relationship type will be deduced by using deductive reasoning algorithm depending upon the obtained relationship types between start person and end person. Thus, input cypher query does not need to use. For Unrelated Graph, if relationship does not exist among them, input cypher query is given to retrieve information of these persons. Based on the acquired information, the final relationship type will be inferred by using deductive reasoning algorithm.

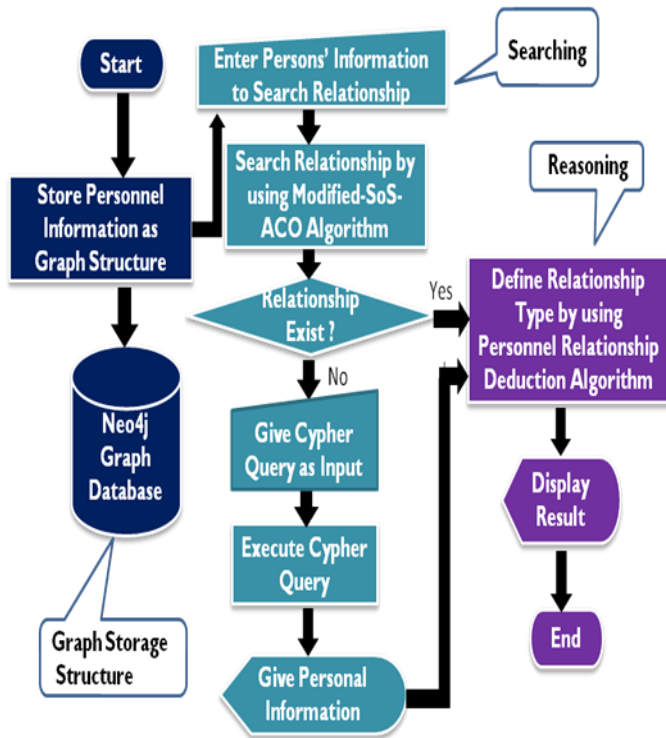


Fig. 1. System Overview.

V. GRAPH STORAGE STRUCTURE

A. Graph Database

A graph database is a database whose specific purpose 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 is any storage system that provides index-free adjacency. This means that every element contains a direct pointer to its adjacent elements and no index lookups are necessary. It can store complex and dynamic relationships of highly connected data like personnel information. There are many advantages of graph database. Flexible data model: no need to declare data types for vertices or edges as opposed to the more constrained table-oriented model of a relational database. 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. Query latency in a graph database is proportional to how much of the graph that choose to explore in a query, and is not proportional to the amount of data stored. Querying connected data from data of any significant size or value. 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 research 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].

B. Graph Data Model

In our proposed framework, personnel information of every member of families are created as graph structure with persons as nodes and links between them as relationships as shown in figure 2. One family and another family may be related or unrelated. Nodes contain properties called key-value pairs. Relationships also contains properties, name and direction [8]. In our graph model, three depth of ascendants and descendants are defined for persons who are parents of the age between 50 and 70. For every person, 28 properties are defined such as name, father_name, mother_name, husband_name, wife_name, brother_name, sister_name, son_name, daughter_name, nrc_no, race, nationality, religion, date_of_birth, gender, marital_status, occupation, department, organization, job_location, phone_no, current_address, permanent_address, native_town, alive/death, hobby, favourite_music and movie. Relationships are created according to the connections between persons.

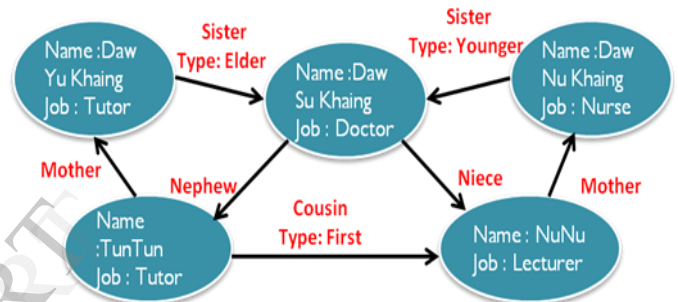


Fig. 2. Graph Data Model.

C. Neo4j Graph Database Storage Structure

Neo4j graph database is a browser based command driven client, like a web-based shell environment. For manipulating Neo4j Graph Database, nodes and relationships are firstly created. Then, graph data can be retrieved by using Cypher query language. Query results are displayed with tabular form and graph visualization form containing nodes and relationships [12]. For creating single family graph as shown in figure 3, five persons nodes with properties and relationships between them are needed to create by using cypher query language in Neo4j graph database. The output of single family graph as graph visualization form is displayed in figure 4.

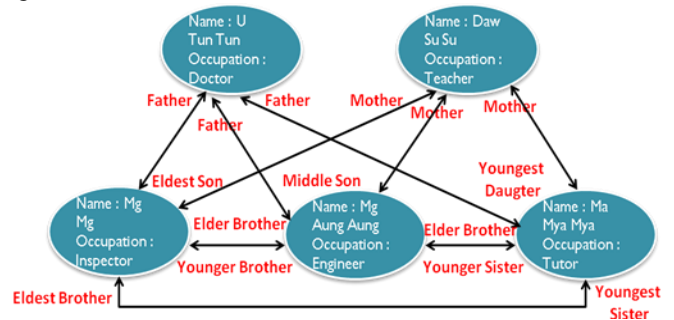


Fig. 3. Creating Single Family Graph.

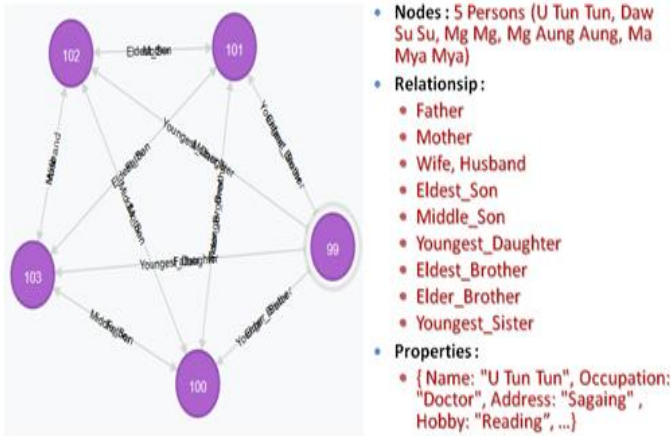


Fig. 4. Creating Single Family Graph with Neo4j Graph Database.

Cypher is a declarative graph query language for updating of the graph store without having to write traversals through the graph structure in code. Cypher focuses on what to retrieve from a graph not how to do it, in contrast to imperative languages like Java, and scripting languages [12]. In this paper, cypher query is used to find the relationship between persons who are not related from separate graphs. For example query, Pattern matching can also be used to make recommendations. Mg Zar Ni Toe is interested in playing football, so his related persons who are also interested in playing football can be searched. Result of Query in Graph Visualization Form and Tabular Form is shown in figure 5.

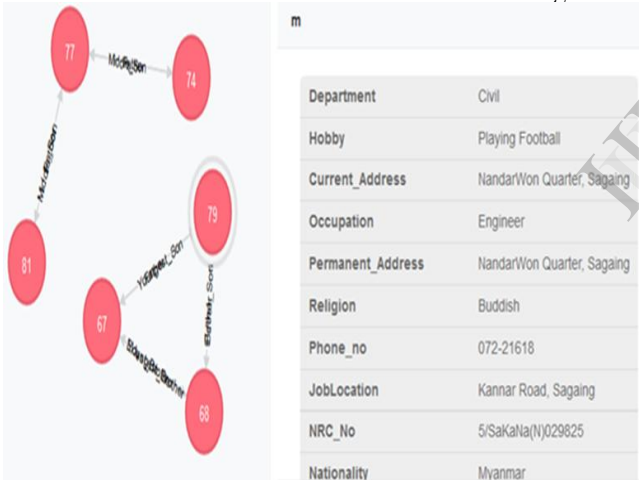


Fig. 5. Cypher Pattern Matching for Recommendation.

VI. GRAPH SEARCHING ALGORITHM

For the relationship between two persons who are connected directly, cypher query can be easily used to retrieve the persons' relations. However, for the relationship between two persons who are not connected directly, searching the persons' relations with cypher query can cause difficulties. If the relationship types among persons are exactly known, cypher query will be used with no trouble. In the graph database with large amount of nodes and relationships, to exactly know the relationship between nodes is not possible. Moreover, when the size of the graphs is more bigger and bigger, the time to search paths in large graphs should be fast. Therefore, searching algorithm related to graph database like Modified-SoS-ACO (Sense of Smell - Ant Colony Optimization) algorithm is applied to find relationship of persons within the few number of processing steps and shortest time.

Modified SoS-ACO Algorithm
 input : personnel information graph database G
 output: relationship types Rt or null
 begin
 1. Formalization of the problem
 2. Selection of food nodes
 3. Diffusion of food odor
 4. Path searches phase
 end

Algorithm 1 Modified SoS-ACO Algorithm

Case Study 1. Related Family Graph

Case study 1 focus on finding the connection between two persons whose relation has already defined by their properties, as shown in figure 6, Mya Mya and Nu Nu.

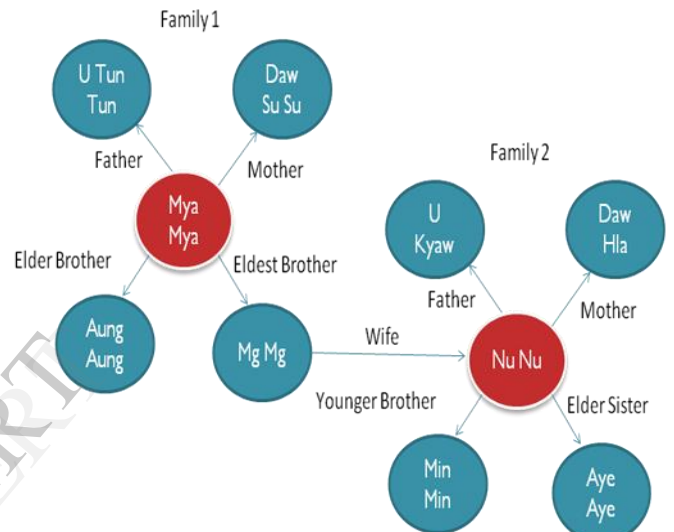


Fig. 6. Case study 1 : finding two persons connection using predefined properties.

Given personnel information graph database, it is represented by a large amount of connected family tree graphs and unrelated graphs that it is defined as $G(N, L)$: $N = \{n_i\}$ is a set of nodes and $L \subseteq N \times N$ is a set of edges. The sample graph in case study 1 has 10 nodes : $N = \{n1(U Tun Tun), n2(Daw Su Su), n3(Mya Mya), n4(Aung Aung), n5(Mg Mg), n6(U Kyaw), n7(Daw Hla), n8(Nu Nu), n9(Min Min), n10(Aye Aye)\}$; and 9 edges : $E = \{e_{1,2}, e_{1,5}, e_{1,8}, e_{1,9}, e_{2,3}, e_{4,11}, e_{5,4}, e_{5,6}, e_{5,10}, e_{9,7}\} = \{Father, Mother, Eldest Brother, Elder Brother, Younger Brother, Husband, Elder Sister\}$. Start Node $n3$ (Mya Mya) and End Node $n8$ (Nu Nu) are selected as food nodes for finding relationship between them. And then, the nodes that are connected from food nodes are traversed (diffusion of food odor) as shown in figure 7. Start node and end node are related via linking node Mg Mg and so relationship exist between start node = Mya Mya and end node = Nu Nu. Relationship Type Eldest Brother is found between Mg Mg and Mya Mya. Relationship Type Husband is also found between Mg Mg and Nu Nu.

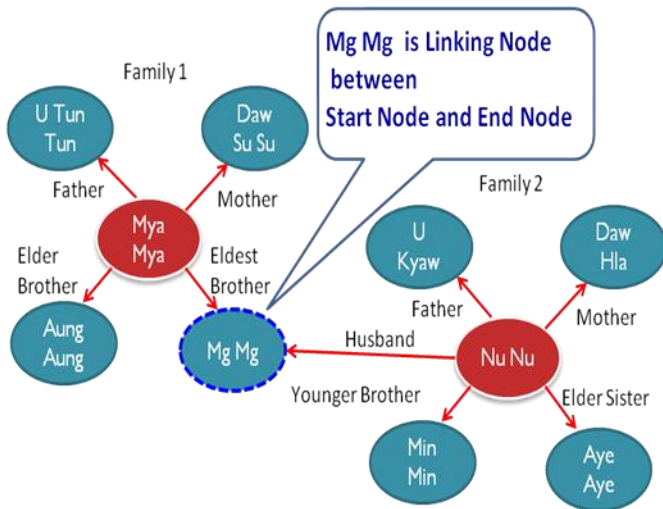


Fig. 7. Diffusion of food odor.

Case study 2. Related Family Graph

Case study 2 focus on finding the connection between two persons whose relation has already defined by their properties, as shown in figure 8, Daw Su Su and Daw Hla.

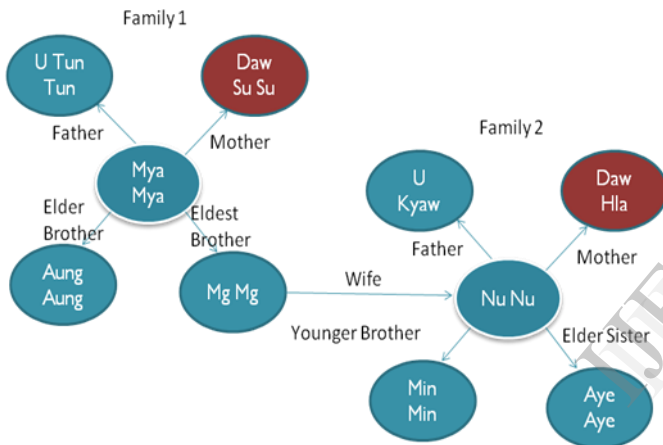


Fig. 8. Case study 2 : finding two persons connection using predefined properties.

The sample graph in case study 2 has 10 nodes : $N = \{n_1(U\ Tun\ Tun), n_2(Daw\ Su\ Su), n_3(Mya\ Mya), n_4(Aung\ Aung), n_5(Mg\ Mg), n_6(U\ Kyaw), n_7(Daw\ Hla), n_8(Nu\ Nu), n_9(Min\ Min), n_{10}(Aye\ Aye)\}$; and 11 edges : $E = \{e_{1,2}, e_{1,5}, e_{1,8}, e_{1,9}, e_{2,3}, e_{4,11}, e_{5,4}, e_{5,6}, e_{5,10}, e_{9,7}, e_{1,4}, e_{2,5}\} = \{Father, Mother, Elder Brother, Eldest Brother, Elder Brother, Younger Brother, Wife, Elder Sister\}$. Start Node $n_2(Daw\ Su\ Su)$ and End Node $n_7(Daw\ Hla)$ are selected as food nodes for finding relationship between them. And then, the nodes that are connected from food nodes are traversed (diffusion of food odor) as shown in figure 9. Start node and end node are related via linking nodes Mg Mg and Nu Nu and so relationship exist between start node = Daw Su Su and end node = Daw Hla. Relationship Type Eldest Son is found between Mg Mg and Daw Su Su. Relationship Type Wife is also found between Nu Nu and Mg Mg. And then, Relationship Type Mother is found between Daw Hla and Nu Nu.

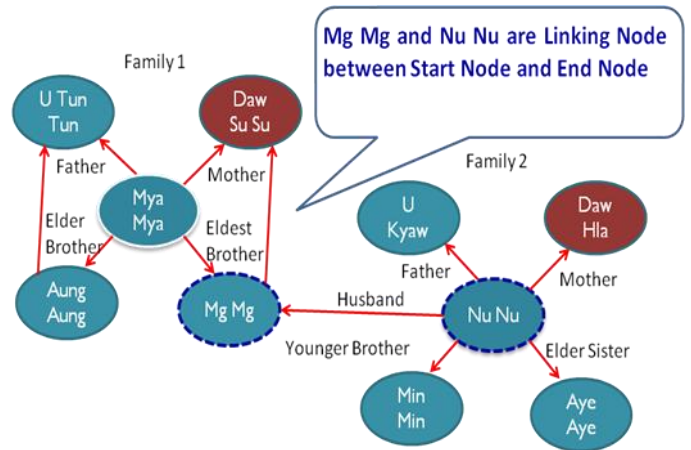


Fig. 9. Diffusion of food odor.

Case Study 3. UnRelated Family Graph

Case study 3 focus on finding the connection between two persons whose relation has already defined by their properties, as shown in figure 10, Daw Khin Khin and Daw Kyi Kyi.

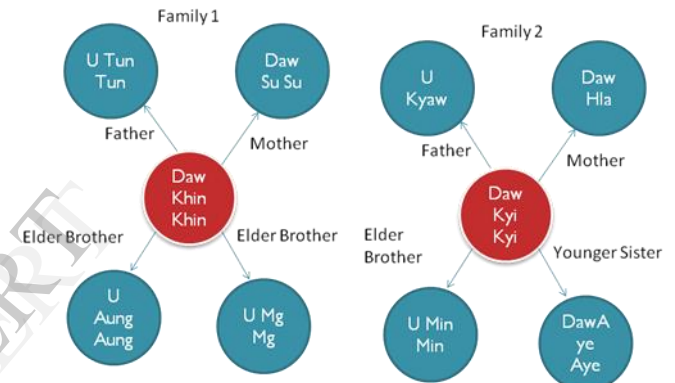


Fig. 10. Case study 3 : finding two persons connection using predefined properties.

The sample graph in case study 3 has 10 nodes : $N = \{n_1(U\ Tun\ Tun), n_2(Daw\ Su\ Su), n_3(Daw\ Khin\ Khin), n_4(U\ Aung\ Aung), n_5(U\ Mg\ Mg), n_6(U\ Kyaw), n_7(Daw\ Hla), n_8(Daw\ Kyi\ Kyi), n_9(U\ Min\ Min), n_{10}(Daw\ Aye\ Aye)\}$ and 8

edges: $E = \{e_{1,2}, e_{1,5}, e_{1,8}, e_{1,9}, e_{2,3}, e_{4,11}, e_{5,4}, e_{5,6}, e_{5,10}, e_{9,7}\} = \{Father, Mother, Elder Brother, Younger Sister\}$. Start Node $n_3(Daw\ Khin\ Khin)$ and End Node $n_8(Daw\ Kyi\ Kyi)$ are selected as Food Nodes for finding relationship between them. And then, the nodes that are connected from food nodes are traversed (diffusion of food odor) as shown in figure 11. Start node and end node are not related and so relationship does not exist between start node = Daw Khin Khin and end node = Daw Kyi Kyi. And so, input cypher query is given to retrieve information of Daw Khin Khin and Daw Kyi Kyi.

Start $p_1 = node(*)$ where $p_1.name = "Daw\ Khin\ Khin"$ return p_1 ;

Start $p_2 = node(*)$ where $p_2.name = "Daw\ Kyi\ Kyi"$ return p_2 ;

According to the above cypher query, personnel information of Daw Khin Khin and Daw Kyi Kyi are acquired as shown in figure 12.

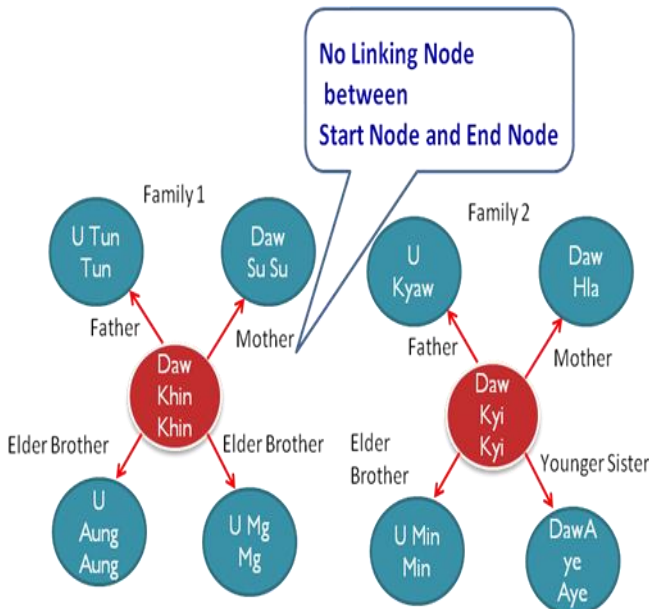


Fig. 11. Diffusion of food odor.

Personal Information of Daw Khin Khin

- Name : Daw Khin Khin
- Age : 40
- Occupation : Lecturer
- Organization : UCSY
- Address : Hlaing Campus, Yangon
- Hobby : Reading

Personal Information of Daw Kyi Kyi

- Name : Daw Kyi Kyi
- Age : 26
- Occupation : Tutor
- Organization : UCSY
- Address : Hlaing Campus, Yangon
- Hobby : Playing Piano

Fig. 12 Personnel information of Daw Khin Khin and Daw Kyi Kyi.

VII. DEDUCTIVE REASONING ALGORITHM

A. Reasoning

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 main types of reasoning such as Deductive Reasoning, Inductive Reasoning, Abductive Reasoning Reductive Reasoning and Fallacious Reasoning.

B. Deductive Reasoning

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. It 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 reached is necessarily true [11].

Personnel Relationship Deduction Algorithm

input : relationship types R_t , personnel info P , a rule database D

output: an specific conclusion C or null

For each rule r in D

if $r \vdash R_t C$ then (if relationship types R_t match with rule r then give conclusion of that rule)

return C

if $r \vdash P C$ then (if personnel info P match with rule r then give conclusion of that rule)

return C

else

return null

end

end

Algorithm 2 Personnel Relationship Deduction Algorithm

Case Study 1. Personnel Relationship Deduction

By Modified SoS-ACO Algorithm's Result, Relationship Type : Eldest Brother is obtained between Mg Mg and Mya Mya. Moreover, Relationship Type : Husband is also obtained between Mg Mg and Nu Nu. Thus, by Personnel Relationship Deduction Algorithm, Let Mg Mg be Person1, Mya Mya be Person2 and Nu Nu be Person3. And so, Person1 is Eldest Brother of Person2 and Person1 is Husband of Person3. After that, by deductive reasoning rule, if Person1 is Eldest Brother of Person2 and Person1 is Husband of Person3 then Person2 is Sister in Law of Person3. Therefore, final result is given out Sister-in-law Relationship between Mya Mya and Nu Nu.

Case Study 2. Personnel Relationship Deduction

By Modified SoS-ACO Algorithm's Result, Relationship Type : Eldest Son is obtained between Mg Mg and Daw Su Su. Moreover, Relationship Type : Wife is obtained between Nu Nu and Mg Mg. Besides, Relationship Type Mother is also obtained between Daw Hla and Nu Nu. Thus, by Personnel Relationship Deduction Algorithm, Let Mg Mg be Person1, Daw Su Su be Person2, Nu Nu be Person3, and Daw Hla be Person4. And so, Person1 is Eldest Son of Person2 and Person3 is Wife of Person1 and Person4 is Mother of Person3. After that, By Deductive Reasoning Rule, if Person1 is Eldest Son of Person2 and Person3 is Wife of Person1 and Person4 is Mother of Person3 then Person2 and Person4 are khami khamet. Therefore, final result is given out Khami Khamet Relationship between Daw Su Su and Daw Hla.

Case Study 3. Personnel Relationship Deduction

By Modified SoS-ACO Algorithm's Result, relationship does not exist between Daw Khin Khin and Daw Kyi Kyi. Thus, personnel information of these two persons are queried with cypher query language. After that, relationship between them can be deduced based on acquired personnel information by using personnel relationship deduction algorithm. Organization and Address of Person1 are same with Organization and Address of Person2 and hence, final result can be infered as Daw Khin Khin and Daw Kyi Kyi work in same organization and live in same address.

VIII. CONCLUSION AND FUTURE WORK

To cap it all, our proposed framework can search the personnel relationship whatever the persons who are connected via one or more intermediate persons like related graphs or separated graphs. With the support of graph database, complex and dynamic relationships of highly connected data like personnel information can easily be created. For data of any significant size or value, it is the best way to represent and query connected data that does not need joins. Modified SoS-ACO algorithm also makes easy to search on large graphs without the need to change the structure of the graph or carry out a complex pre-processing steps. Due to the effectiveness of personnel relationship deduction algorithm, any relationship among persons can simply and accurately be inferred. The overall goal of this paper is to search different kinds of relationship between persons that are stored in graph database. Modified-SoS-ACO algorithm is exploited to provide the desired result within the short processing time and few numbers of steps. Deduction Algorithm is also used to deduce the final relationship between persons. For doing predictive analysis of personnel relationship, this paper is shown some experiments on the Census data and including facts for creating NRC card, personal profile and other additional information such as hobby, favorite music and movies.

For further development, our proposed system can be extended to test the experiments over dynamic graphs. Moreover, this system is implemented by using homogeneous database such as single graph database. Therefore, this system can be extended to implement by using heterogeneous databases like relational, nosql and graph database. Furthermore, the different query processing time is wanted to compare based on retrieving related data from those databases by using their relevant query system and find out which query process can produce the optimal running time.

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/Deductive_Reasoning
- [12] <http://www.neo4j.org>