Journal of Engineering and Applied Sciences 12 (Special Issue 7): 8082-8087, 2017
ISSN: 1816-949X

The Formalization of Tree Structure for the Genealogy System

Noraida Haji Ali and Masita Masila Abdul Jalil
School of Informatics and Applied Mathematics,
Universiti Malaysia Terengganu, Terengganu, Malaysia

Abstract: Genealogy is a piece of activity relating to the research of a descendant of a family or a person. At present, genealogy research activity especially in the context of Malay society is carried out manually. This study describes the definition of 23 genealogy rules created to enable the generation of the Malay family relations by a genealogy system and the detection of the family relation using the tree concept or tree technique. These rules would be used in the genealogy database and are formalized to show the structure of nodes involved in determining a family tree.

Key words: Genealogy, formalization, family tree, tree structure, relations, Malaysia

INTRODUCTION

Genealogy is a piece of activity relating to the research of a descendant of a family or a person. Earlier researches on genealogy mainly focused on finding the descendant of nobles and rulers but now, more people are conducting genealogy research to find and maintain their own family tree (Yakel, 2004).

Many genealogy software tools are available today to assist genealogist to systematically collect, store, sort and display genealogical data. Most programs also enable users to generate charts to represent the family tree. In the Western countries, especially in the Christians communities, some of these systematic methods and tools have been widely used in facilitating management and manipulation of family records. In addition, a standard file format for genealogy activity, GEDCOM (Genealogy Data COMmunication) has also been developed. GEDCOM is used by a number of genealogy software for data interchange between the different software. Some of the software is distributed freely on the internet. Unfortunately, all of the software is not suitable for the Malay race. One of the main characters that are not suited is the name structure which is different compared to our name structure. Furthermore, some of the definitions of family relationships are different from those being used or adopted by the Malays.

Researches on genealogy could be grouped into two broad categories. The first category focuses on means to enhance features of genealogy tools. Wesson *et al.* (2004) for instance have designed a zoomable interface to facilitate dynamic exploration and browsing of family trees in their ZoomTree tool. Another genealogy tool, Phlips has incorporated expert system technology that utilizes rules to assist genealogists with their searches (Yang *et al*., 2007). Nuanmeesir *et al*. (2009) have also looked into problems of genealogy information search and have applied Parent Bidirectional Breadth Algorithm (PBBA) to enhance the search performance.

Another category of research looks into alternatives in representing relationships in the family tree such as by Katsaros *et al*. (2005) and Nishimura *et al*. (2002). XML technology has also been utilized to help create metadata of genealogy information (Yeh and Chen, 2003). On the other hand, Jiang and Dong (2008) proposed the use of ontology to represent family relations. Nodes and edges are used to represent a member of the family and the relation between any two members, respectively.

The study basically focuses on rules generation for Malay family relations. Unlike the Western families, the Malays, for example, do not carry their surnames or family names after they got married. Rather, the husband will still use his first name and so as the wife. Hence, the rules that have been adopted by most of the available genealogy tools are to some extents inaccurate to be applied to our family structure. The newly generated rules try to cater for these differences and can be deployed in any newly designed genealogy system.

MATERIALS AND METHODS

The genealogy rules: Genealogy rules module is the most important part of this system. This module consists of 23 different rules or algorithm of programming codes that specify family relations. These rules are designed based on the Malay family relations which depend on two

Corresponding Author: Noraida Haji Ali, School of Informatics and Applied Mathematics, Universiti Malaysia Terengganu, Terengganu, Malaysia

factors marriage and descendant. Therefore, the rules or codes are run based on the Identification number (IC) and the gender or sex of each person in the record.

Components of genealogy rules: Each of these rules comprises of 3 major components:

- Component 1: functioning as the information seeker. It will search the record needed by the system during the process
- Component 2: functioning as the 'random access memory' for the system. It will store or hold the entire ID found by component 1
- Component 3: it compares and checks the similarity between two IDs

These components work together as an integrated subsystem to perform the corresponding process. Although, each of the rules has all the components, it differs in combination and structure of their sequences. The system processes the information supplied by the 'Identification card number' (referred to as 'ID') and 'sex' in the database. This information is stored in the ID, ID_Father, ID_Mother, ID_Couple and sex attributes.

The genealogy rules: The process used to decide on an accurate family relation can be viewed as the process done by the system to complete what is called the 'relationship statement' as shown:

Person 1 is a 'family relation' to person 2

Person 1 and 2 are the corresponding persons or family members whose type of relationships to be discovered by the system while the family relation is the relationship that the system is going to complete. As an example, let say a user wanted to know what the relationship between Ali and Ahmad is. The system will place both names in the person 1 and 2 and generate a result as follow:

Ali is a 'family relation' to Ahmad

Next, the system will fill in the 'family relation' with a correct 'word' by applying the genealogy rules that are built in the system. This process involves all the 3 components in each rule. It will search all the information one by one to satisfy the rules. If succeed for example, the system found that Ali is the grandfather, the system will complete the sentence or 'relationship statement' as:

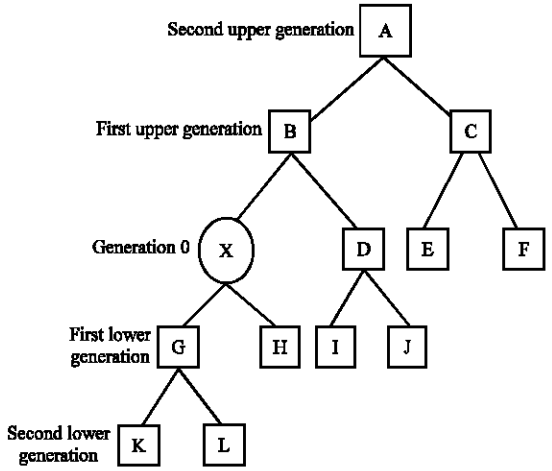


Fig. 1: Basic family tree structure

Ali is a grandfather to Ahmad

Application of the tree structure: Figure 1 shows the basic family tree structure which has been used to design the rules for each family relation covering two upper and two lower levels of the family hierarchy.

Figure 1 clearly shows 5 lines indicating 5 levels of generations. There are also many nodes labeled with a different letters and are placed in different levels or generations. While the node with label 'X', shows the person 1 (as stated before), person 2 can be anybody or any nodes in the 'tree' based on what has been entered by the user. Below is the major algorithm for the rules:

- Firstly, the system will search for the ID of person 1 by traveling and searching the entire database
- When found, the system will continue searching for the ID of person 2
- Next, the system will place them in a certain nodes in the family tree
- Then, the system will compare and decide on the type of relations that can be initiated between these two individuals. This process is done by using the family tree

For this system, we have 23 family relations as used by the Malays. The family relations which can be generated by the system. Examples for each family relation are given based on nodes in Fig. 1.

List of family relations:

- Grandfather on father's side, e.g., 'A' is a grandfather on father's side of 'X'

- Grandfather on mother's side, e.g., 'A' is a grandfather on mother's side of 'X'
- Grandmother on father's side, e.g., 'A' is a grandmother on father's side of 'X'
- Grandmother on mother's side, e.g., 'A' is a grandmother on mother's side of 'X'
- Uncle on father's side, e.g., 'C' is an uncle on father's side to 'X'
- Uncle on mother's side, e.g., 'C' is an uncle on mother's side to 'X'
- Aunt on father's side, e.g., 'C' is an aunt on father's side to 'X'
- Aunt on mother's side, e.g., 'C' is an aunt on mother's side to 'X'
- Cousin on father's side, e.g., 'E' is a cousin on father's side to 'X'
- Cousin on mother's side, e.g., 'E' is a cousin on mother's side to 'X'
- Father, e.g., 'B' is a father of 'X'
- Mother, e.g., 'B' is a mother of 'X'
- Sibling, e.g., 'D' is a sibling of 'X'
- Child (son/daughter), e.g., 'G' is a child of 'X'
- Niece/nephew, e.g., 'I' is a niece/nephew of 'X'
- Father-in-law, e.g., 'X' is a father-in-law of 'M'
- Mother-in-law, e.g., 'X' is a mother-in-law of 'M'
- Son/daughter-in-law, e.g., 'M' is a son or daughter-in-law of 'X'
- In-law, e.g., 'M' is an in-law of 'H'
- Husband or wife of an in-law, e.g., 'M' is a husband or wife of an in-law of 'N'
- Husband, e.g., 'M' is a husband of 'G'
- Wife, e.g., 'G' is a wife of 'M'
- Grandchild, e.g., 'K' is a grandchild of 'X'

Example for family relation 'grandchild': An example that illustrates a 'Grandchild' family relation is given in Fig. 2. As shown above, the lines connecting nodes 'X' with nodes 'B' and 'A' are highlighted in bold to indicate their generation sequence. This study will discuss how the system recognize the family relation that exists between 'X' and 'A'. That is 'X' is a 'Grandchild" to 'A'. To show how such recognition is established for this family relation, a numbering label is used as a symbolic programming sequence. By referring to Fig. 2, again there are 3 circled numbers which are 1-3, respectively.

The following steps are program sequences to recognize family relation between the nodes 'X' and 'A': genealogy rules will define ID of individuals as entered by users. If this ID is found in the database, 'X' node will be labeled as number 1 (Fig. 3). Otherwise, the program will stop here.

If that ID exists, the program will read the record ID_BAPA from the database for 'X' node and continue to

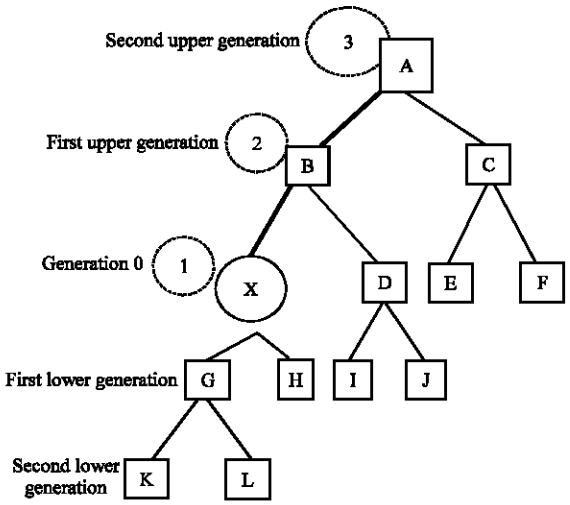


Fig. 2: Family tree structure

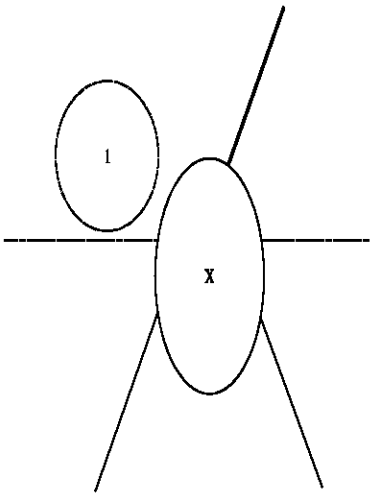


Fig. 3: 'X' node

seek for ID which have the same value with the value of ID_BAPA for 'X' node. If the search is successful that ID will be labeled as number 2 in this case, it will refer to 'B' node (Fig. 4).

Then, the program will repeat the second step but this time it will seek for ID_BAPA for 'B' node. If this ID_BAPA is in the database that ID will be labeled as number 3 (Fig. 5).

Finally, the program will make comparison between the ID of 'A' node with Individual's ID. If these values agreed, the genealogy rule will define the right family relation for this relationship. The system will then generate statement for this relationship that is:

X is a 'grandchild' to 'A'

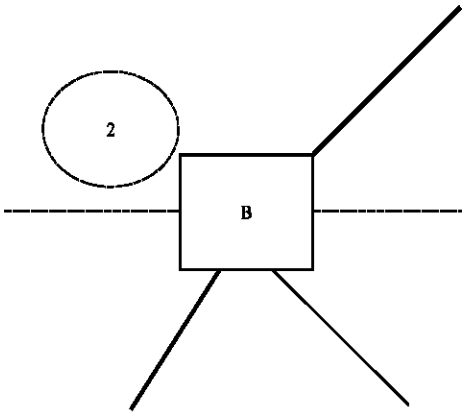


Fig. 4: 'B' node

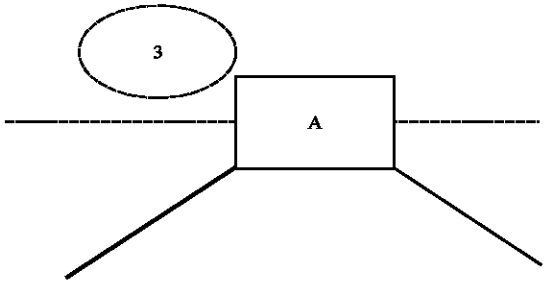


Fig. 5: 'A' node

RESULTS AND DISCUSSION

Formalization of the tree structure: A database is required to keep track of genealogical relationships between people (family trees). It would be possible to represent the required relationships (parent, grandparent, aunt, cousin, etc.) separately but this would limit the number of relationships, increase the complexity of the specifications and would make it necessary to carry out extensive integrity checks every time the database is updated. We will represent this database using formal specification to define operations to output any required relationships. The most fundamental genealogical relationship is that of parent to child and this, together with the sexes of all individuals in the database will be enough to enable us to specify all the operation we require.

We adapted formal representation given by Currie, (1999) to fulfill this research requirement. This suggests the following types:

$$[\text{Person}] \text{ the set of all people}$$

$$\text{Gender} :: = \text{Male} | \text{Female}$$

The parent/child relationship can be represented by the relation:

$$\text{Parent: person} \leftrightarrow \text{person}$$

Where:

$$H(x,y) == x \mapsto y \in \text{parent}$$

relation represents the information that x is parent of y. We can also represent this relationship by the following nodes:

$$\text{Sex}_{y}^{x}\text{: person} \rightarrow \text{gender}$$

Where:

$$\text{Dom sex} \in \text{person}$$

is the set of all people in the database while:

$$G(s, t) == \text{dom}(s \mapsto t) \in \text{Person and t: Gender}$$

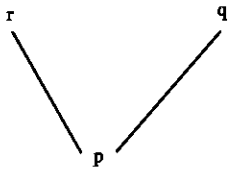
The sexes of all the people in the database can be represented by the function. Represents the information that s has gender t. For example:

$$H(x,y) \rightarrow G(x, \text{male}) \rightarrow \text{father}(x)$$

represents that x is father of y because of x is parent of y and x is male where x person. The final restriction is that anyone in the database can have a maximum of two parents and if they:

$$\forall p, q, r\text{: Person } \{q \mapsto p, r \mapsto p\} \cdot Q \neq r \cdot \text{sex } q \neq \text{sex } r$$

have two, the parents must be of opposite sexes! If we draw the node for this representation, it looks like this. In addition, sex q≠sex r to ensure the person of r or q is a parent of p. We can set the person, r is male and the person, q is female and can represent these relations by:

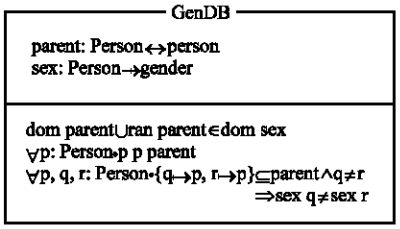


Alternatively, we can represent it as F(x) == x is female which represents the information that x is a female. The above statement can be represented by:

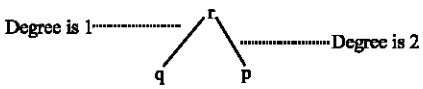
$$H(q,p) \wedge H(r,p) \leftrightarrow F(r) \wedge \neg F(r)$$

where, q and r are parents of p, q is female (mother) and r is not female that's means r is a male (father).

Algorithm 1; The state scheme is as follows:



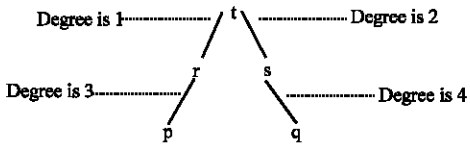
We will further stipulate that the set contains only the common ancestors of minimum degree where the degree refers to the number of steps in the path up and/or down the family tree between the two people. For example, if ps and qs are siblingss, the degree is 2. We can look at this tree structure:



Where:

$$\{\forall p, q, r: person \cdot \{r \mapsto p \wedge r \mapsto q\}\}$$

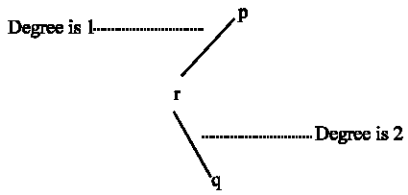
If ps and qs are first cousins, the degree is 4:



Where:

$$\{\forall r, p: person \cdot \forall q: person \cdot \{p \mapsto r \wedge r \mapsto q\}\}$$

If ps is the grandparent of qs, the degree is 2:

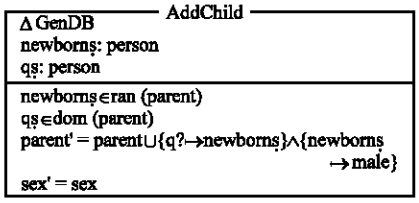


Where:

$$\left\{\begin{array}{l}\forall r, s, t: person \cdot \forall p, q: person \cdot \\ \{r \mapsto p \wedge s \mapsto q\} \wedge \{t \mapsto r \wedge t \mapsto s\}\end{array}\right\}$$

We need an operation scheme to add a new child for parents that already existed in database. This operation must return the set cas! containing the common ancestors of two people, say newborns and qs:

Algorithm 2; add child:



The precondition is that all the people involved are in the database:

$$\{q\varsigma, q\varsigma\} \cup cas! \subseteq dom\, sex$$

The post-condition characterizes elements of the set of common ancestors of minimum degree as people for whom there are multiple compositions of parent which map both ps and qs to them and furthermore, there are no other common ancestors with degree smaller than that of members of this set.

Since, newborns represents a new child of the parent, q therefore newborn is not regarded as a member yet while the parent, q, must already exist in database.

After we add a new record, newborn s in the database, the list of records will be appended with new record. We can also give the gender relation for the new record.

CONCLUSION

This study describes the definition of 23 genealogy rules created to enable the generation of the Malay family relations. These rules would be used in the genealogy database and are formalized to show the structure of nodes involved in determining a family tree. An expert system specifically attributed to the Malay genealogy field could directly benefit from these rules.

REFERENCES

Currie, E., 1999. The Essence of Z. Prentice Hall Europe, UK., ISBN-13: 9780137498390, Pages: 187.

Jiang, Y. and H. Dong, 2008. Ontology based knowledge modeling of Chinese genealogical record. Proceedings of the IEEE International Workshop on Semantic Computing and Systems, July 14-15, 2008, Huangshan, China, pp: 33-34.

Katsaros, D., A. Nanopoulos and Y. Manolopoulos, 2005. Fast mining of frequent tree structures by hashing and indexing. Inform. Software Technol., 47: 129-140.

Nishimura, N., P. Ragde and D.M. Thilikos, 2002. On graph powers for leaf-labeled trees. J. Algorithms, 42: 69-108.

Nuanmeesir, S., C. Baitiang and P. Meesad, 2009. Genealogical information search by using parent bidirectional breadth algorithm and rule based relationship. Int. J. Comput. Sci. Inform. Secur., 6: 1-6.

Wesson, J., M.C. du Plessis and C. Oosthuizen, 2004. A zoomtree interface for searching genealogical information. Proceedings of the 3rd International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa, November 3-5, 2004, Cape Town, South Africa, pp: 131-136.

Yakel, E., 2004. Seeking information, seeking connections, seeking meaning: Genealogists and family historians. Inform. Res., Vol. 10.

Yang, H.H., D. Forrester and D. Harris, 2007. A PHLIPS-based expert system for genealogy search. Proceedings of the IEEE Southeast Conference, March 22-25, 2007, Richmond, VA., USA., pp: 165-170.

Yeh, J.H. and C.C. Chen, 2003. Knowledge management in a Chinese genealogy information system. Proceedings of the 6th International Conference on Asian Digital Libraries, December 8-12, 2003, Kuala Lumpur, Malaysia, pp: 427-431.