Social Network Recommendations for Friends with Neo4j Graph Database

Pham Thi Thu Thuy[1], Nguyen Thi Thai Thanh[2], Hwa Soo Kim[3]

[1] Nha Trang University, Nha Trang, Vietnam

[2] Van Ninh Vocational School, Khanh Hoa province, Vietnam

[3] Graduate School of Information and Communication Technology, AJOU University, Korea

Correspondence: Pham Thi Thu Thuy, thuthuy@ntu.edu.vn

Received: 02/28/2023, revised: 11/09/2023, accepted: 04/10/2023

Digital Object Identifier:: 10.32913/mic-ict-research-vn.v2023.n2.1239

Abstract: In recent years, along with the development of the internet, the number of social network users is increasing day by day. Through social networks, users can share, exchange information or make friends with each other. However, with new relationships, users often have a need to assess the credibility of a new friend before making friends on social networks. This paper proposes a friend recommendation system on social networks based on how to calculate the reliability between users. The recommendation system is implemented using the Neo4j graph database. The results of the truth algorithm proposed in this paper are higher than other similar algorithms.

Keywords: *Social Network, Graph Database, Neo4j, Recommendation, Truth algorithm*

Tiêu đề: **Ứng dụng cơ sở dữ liệu đồ thị Neo4j xây dựng hệ thống khuyến nghị kết bạn trên mạng xã hội**

Tóm tắt: Những năm gần đây, cùng với sự phát triển của internet, số lượng người dùng mạng xã hội ngày càng tăng. Thông qua mạng xã hội, người dùng có thể chia sẻ, trao đổi thông tin hoặc kết bạn với nhau. Tuy nhiên, với những mối quan hệ mới, người dùng thường có nhu cầu đánh giá độ tin cậy của một người bạn mới trước khi kết bạn trên mạng xã hội. Bài báo đề xuất hệ thống gợi ý kết bạn trên mạng xã hội dựa trên cách tính toán độ tin cậy giữa những người dùng. Hệ thống đề xuất được triển khai bằng cơ sở dữ liệu đồ thị Neo4j. Kết quả của thuật toán xác thực đề xuất trong bài báo này cao hơn các thuật toán tương tự khác.

Từ khóa: *Mạng xã hội, Cơ sở dữ liệu đồ thị,Neo4j, Khuyến nghị, Thuật toán tính độ tin cậy*

I. INTRODUCTION

In recent years, computers have brought tremendous opportunities for the development of society. We use computers every day; they not only become part of our daily lives, but also help define our personality through our social network profiles, virtual activities, our behavior in discussions, etc. Changes in human behavior inevitably affect the technical and functional requirements of computers from applications. The Internet has become a necessity; this places a huge demand on web applications. Social networking site Facebook has launched Web 3.0, which is not only a modern network but also a network of intelligent, personalized user and multimedia content.

Through social networks, users can interact, share articles, post photos and videos, and help users connect with people currently living in different geographical regions or allow users to make friends and looking for friends. The main components in a social network include nodes and links. A node is an entity or a collection of entities such as individuals, businesses or organizations also called nodes. A link is a relationship between nodes. In a network in its simplest form, a social network is an arrangement of related links between nodes. A social network can be represented by a diagram in which the linked nodes are shown as straight lines.

Therefore, social networks create a social graph of users, including users and their friendship relationships. This social graph is very rich, not just the relationship of one or two friends to each other but the relationship between many friends (for example, friends of a friend). In that vast relationship, sharing and exchanging information or making new friends,... are mostly based on mutual trust. Therefore, just like in life, trust among users in social networks is also concerned and has been mentioned by many researches in computer science in recent years.

On the other hand, graph databases are used in data with many relationships such as social networks, recommender systems, network management, geospatial, etc. Strengths of graph databases is the ability to store relationships and query on those relationships. A graph database is a type of NoSQL [18] database where data is organized and stored in the form of nodes, edges, and attributes. Unlike the relational database model, the graph database focuses on describing the relationships of the data, which makes it well supported for data types with many complex relationships. complexity and queries related to these relationships.

Currently there are graph databases being used such as: Neo4j [1, 2], RedisGraph [3]; OrientDB [4]. This study chooses Neo4j database to represent social network data. Neo4j is an open source graph database that is a NoSQL database, built in Java and Scala, sponsored by Neo Technology Corporation. The project began to develop in 2003 and became public since 2007. Currently, Neo4j has been used by thousands of companies and organizations in various fields, including network management, software analysis, scientific research, routing, and social networking [17]. Neo4j is well suited for performing association prediction problems, as well as for big data management. Neo4j is mainly designed to capture relationships between items of information, the basic organizing principle of Neo4j is to store information as a network of relationships. The network contains nodes, the nodes are connected through relationships, the nodes are called vertices and the relationships are called edges. Neo4j is designed to handle big data, highly interconnected data, supports multiple platforms, and has the Cypher [5] query language that queries graphs very quickly.

Cypher is a declarative query language similar to SQL, queries are constructed using different clauses. Cypher is designed to be simple, powerful and very convenient when working with Neo4j. Cypher's c structure is designed and built based on the English language, neat, easy to understand, convenient for language manipulators to make reading and writing queries easy and suitable for developers. programmers and database query specialists.

The objective of this study is to learn how to retrieve and represent social network data, specifically Facebook, in the form of a Neo4j database, and then evaluate the trustworthiness of each user account on the Internet. Facebook through their activities. On the basis of the installation results, we install the algorithm that has been previously studied by the previous researchers and propose a new reliability calculation algorithm to improve the reliability calculation results among users on social networks.

The friend recommendation problem is often used in social networks. The basic approach is based on referrals from your friends, the same group, the same events, the same interests... there are many connections to each other, so the use of a graph database to store data exploits many of the strengths of the graph data model.

This study aims to reconstruct the database of users on Facebook as a Neo4j graph database, and install algorithms to determine the trust between users such as Adamic Adar, Common Neighbors and Preferential Attachment, thereby suggesting friend recommendations.

In addition, we propose a new algorithm related to calculating trust among users based on their social relationships. The results of the study have formed an application that recommends making friends based on specific relationships such as: the relationship between friends on each level or the relationship about where you live, or where you work.

II. RELATED WORK

This section presents some commonly used algorithms in social network friend recommendation systems. The most commonly used algorithm is Common Neighbors (CN) [6]. The common neighbor algorithm captures the idea that two strangers who have a mutual friend are more likely to be referred than those who have no mutual friends. The similarity between nodes a and b is calculated as the number of common neighbors between a and b. The greater the number of shared nodes, the higher the similarity between a and b.

$$CN(a,b) = |N(a) \cup N(b)| \quad (1)$$

The similarity between nodes a1 and b1 using CN similarity measurement can be calculated as in equation (2):

$$CN(a1,b1) = |N(a1) \cup N(b1)| = |\{1,2\} \cup \{1,5\}| = 1 \quad (2)$$

Another commonly measure is Adamic Adar (AA) [7] as in the equation 3. It is a measure that is used to calculate the closeness of two nodes based on their common neighbor. The similarity between nodes a and b is calculated as the sum of the reciprocals of the logarithms of each common neighbor z between a and b, the more common nodes at a low level, the higher the similarity between nodes. This is a weighted version of the CN similarity measure.

$$AA(a,b) = \sum_{z \in Na_1 \cap N(b_1)} \frac{1}{log(x_z)} \quad (3)$$

The similarity between nodes a1 and b1 using the AA measure can be calculated as in equation (4):

$$AA(a,b) = \sum_{z \in N a_1 \cap N(b_1)} \frac{1}{log(x_z)} = \frac{1}{log(x_1)} = \frac{1}{log(3)} \quad (4)$$

N(z) is the set of nodes adjacent to z. A value of 0 indicates that the two nodes are not close to each other, while a higher value indicates that the nodes are closer together.

Another algorithm mentioned in the similarity assessment system on social networks is Preferential Attachment (PA) [8] model. In this model, the more connected a node is, the more likely it is to receive new links. A value of 0 indicates that the two nodes are not close to each other, a higher value indicates that the nodes are closer together. The similarity between nodes a and b is calculated as the product of the degrees of nodes a and b. The higher the degree of both nodes, the higher the similarity between a and b, as shown in the equation (5). Where xa is the number of relationship levels of User 1; xb is the number of relationship levels of User 2.

$$PA(a,b) = x_a * x_b \quad (5)$$

In the previous work, the lead author proposed distance-based reliability measurement [9] based on the assumption: the closer a node is to the source node in a trusted network, the more reliable it is. The measurement is given as follows: For power button E; E predicts the reliability of all remaining nodes in the system based on the shortest distance from E to those nodes. Suppose we call d the maximum threshold over which the confidence can propagate. For a node A whose shortest distance from E to A is n(n ≤ d), the reliability of A with E is calculated by the following formula:

$$Trust = \frac{d-n+1}{d} \quad (6)$$

Value of trust = 0 for nodes with distance greater than d. For example, consider the graph in Figure 1.

According to the Figure 1, let's say d=2; Trust (E,A)=1; Trust(E,B)=Trust(E,C)=(2-2+1)/2=1/2;
Trust(E,G)=Trust(E,D)=Trust(E,H)=0

Besides, there are many studies related to link prediction [11–17]. There is an interplay between every pair of different types of relationships. This effect can vary

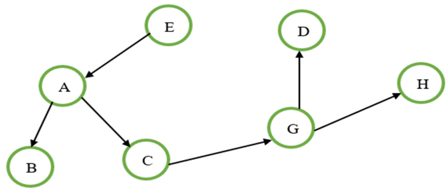

Figure 1. The graph illustrates the example of calculating the distance measure

for different couples. For example, students are more likely to form friendships with friends of classmates than with friends of relatives. The "classmate" relationship has a greater impact on the "friend" relationship than the "relative" relationship. In order to predict potential connections in such multi-relational social networks, the correlation and influence between different relations should be considered comprehensively. Most existing relation prediction algorithms are not suitable for multi-relational networks because they only observe one kind of relation and ignore the overall structure and influence among different relations. An intuitive way to predict hidden or future relations in a multi-relational network is to merge multiple relations in the network into a single relation, and perform prediction of potential relations based on this merged relation. However, with this simple approach, we may miss too much topological information to get accurate prediction results. Therefore, we should develop an effective method to utilize multi-relational information in social networks to improve the accuracy of prediction results.

In this paper, we represent the data of users on Facebook through the Neo4j graph database, apply the algorithms to calculate the reliability between users proposed by previous studies, and at the same time recommend the new algorithm to improve the truth measurement. Detail, we install the above 03 algorithms (including CN, AA and PA), and propose a new algorithm to improve and increase accuracy.

III. OUR PROPOSAL: TRUTH CALCULATION ALGORITHM BASED ON RELATIONSHIPS (DT)

In fact, there are different ways to assess trust in social networks based on direct trust relationships between users, assessment based on reputation of network members, and calculation based on trustworthiness. on trust ratings and there are many types of relationships that connect visitors to each other in a social network.

This paper introduces a new method aimed at calculating trust in social networks, which considers user relationships at each level. The calculated confidence values are in the range 0-1.

Table I
RELIAILITY TABLE BASED ON RELATIONSHIP LEVEL

No	Type of relationships	Trust value
1	Friend	1
2	Your friend	0.8
3	Friend of your friend	0.6
4	Shared workplace	0.4
5	Sharing a place to live	0.2
6	No relationship	0

We construct a direct trust calculation between user v and user v', DT(v; v') as a sorted real value in a unit interval. First, if DT(v; v') = 0, then the confidence v has in v' is minimal, i.e. v is not completely reliable for v'. Conversely, if DT(v; v') = 1, this means that v is completely reliable with v'.

One approach to calculating reliability among users of direct connections is to consider the nature of their interactions based on friendships or co-worker relationships.

Friend: is a type of relationship defined by being a direct relation because v and v' are directly connected through an edge a→a'.

We define the function of a friendship as F:V x R x V → [0,1], this function is used to calculate the confidence ft in the interval [0; 1] between two vertices (two users) depending on their friendship. The formula F(v, r, v') = ft is used as a basis for user v to assign a confidence level to his friends v'.

The input and output parameters of the proposed algorithm are as follows:

Input:

1. U is User
2. d is the level of the algorithm (d > 0)
3. F0 is the set of direct friends of User
4. F1 is the set of direct friends of F.
5. Fd is the set of friends of User according to level d.
6. L is a collection of people living in the same city.
7. W is the set of people who do the workplace.
8. R(m,n) is the resulting 2-dimensional matrix where m is User n is the confidence level.

Output: Truth value, friend recommendation

The pseudo-algorithm code is as in the Figure 2.

Algorithm explanation: The foreach statement line number {2} is executed n times, where n is the number of elements in the set Fd, in each iteration each comparison statement is the lines {3}, {5}, {7}, {9}, {11}, the assignment statements are {4}, {6}, {8},{10},{12}.

```
Begin                          {1}
    foreach f in Fd do:        {2}
       if (f in F0) {          {3}
           R[f, 1]             {4}
       } else if (f in F1) {   {5}
           R[f, 0.8]           {6}
       } else if (f in F2) {   {7}
           R[f, 0.6]           {8}
      } else if (f in W) {     {9}
           R[f, 0.4]           {10}
      } else if (f in L) {     {11}
           R[f, 0.2]           {12}
      }                        {13}
   return R                    {14}
end                            {15}
```

Figure 2. Pseudocode of the proposed algorithm

IV. EXPERIMENT RESULTS

To conduct the simulation of algorithms, we use a computer with Intel Core i5-2430M CPU 2.40GHZ, Ram: 08GB, Windows 10 Pro-64bit operating system. Neo4j Desktop Graph Database version 1.5.7.

To get data on Facebook, we go to https://developers.facebook.com/apps/ and press the "Create application" button. This is the step to create the Facebook application. Any user account that accepts to link with this application will have access and share the information requested by the application (here is information about friends). We then access the above link as a Consumer to select and add the "user_friends" permission and other permissions if necessary. Then we click "Generate Access Token" to get to the application permission screen. We then use the Graph API [10] engine to collect Facebook social network data as illustrative data for algorithms. Using Graph API, we collected 209 users with the following information: Name, MaUID, Gioitinh, Job, Diachi, City, Work. The data consists of 07 columns, 210 lines and 16.1KB in size, in csv file format.

After having user data on Facebook, we create nodes from users and create friendship from these users. We have created 257 Nodes, 615 Relationships and 3 Labels. After creating the relationship, we look for the relationship based on the edges to suggest friend recommendations, and calculate the reliability on that recommendation to help the user to make a right decision.

In this study, we are interested in factors that increase the probability of Users making friends with each other such as: Potential triangle relationship between Users; Users share a specific community such as: living in the same place; same workplace.

Collected data and additional data installed on Neo4j, shown in Figure 3.

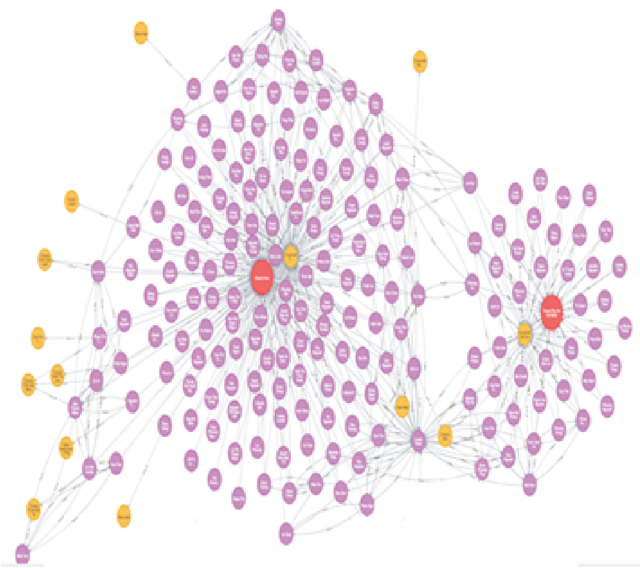

Figure 3. Representation of data collected on Neo4j

In Figure 3, the purple nodes represent the user's nodes on Facebook, the yellow nodes represent the user's workplace, and the red nodes represent the user's workplace.

We install four friend recommendation models, namely: similarity between p1 and p2 nodes by Adamic Adar (AA), common neighbor (CN), priority connection with Preferential Attachment (PA), relationship-based trust proposed by us (DT).

The query to calculate the similarity between node p1 and p2 according to the AA algorithm is as follows:

gds.alpha.linkprediction.adamicAdar(p1, p2) as score;";
System.out.println(queryRaw);
Query query = new Query(queryRaw);
Result result = session.run(query);
return result.single().get("score").asDouble();
} catch (NoSuchRecordException e) {
return 0;

For the CN algorithm, we replace the first line of the above query with the following statement:

gds.alpha.linkprediction.commonNeighbors(p1, p2) AS score;";

Similarly, for the PA algorithm, we replace the first line of the above query with the following statement:

gds.alpha.linkprediction.preferentialAttachment(p1, p2,) as score;";

The software interface of the resulting reliability calculation and friend recommendation for the user is shown in Figure 4. In this figure, when we select any user, the system will list all users in the database that have at least relation with the selected user, for example: Friend, Friend of Friend, Friend of Friend of Friend , where to live, where to work. From there the user can make the decision to make friends if necessary.

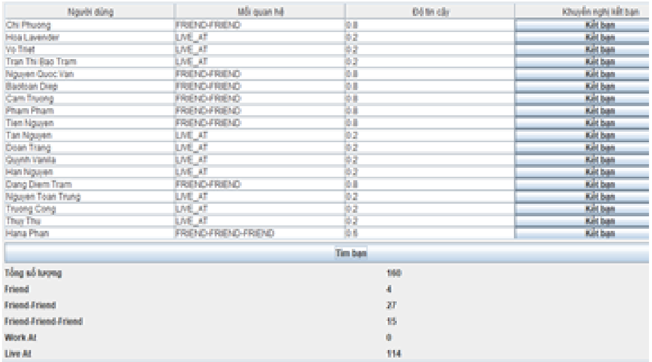

Figure 4. Reliability results and friend recommendations

The matching reliability results on the relationships are shown in Figure 5. Figure 5 shows the matching confidence result of a user named Quang Vinh. Based on the results in Figure 4, we see that our proposed algorithm (DT) always returns the highest matching rate on relationships. The PA algorithm gives the lowest results among the installed algorithms.

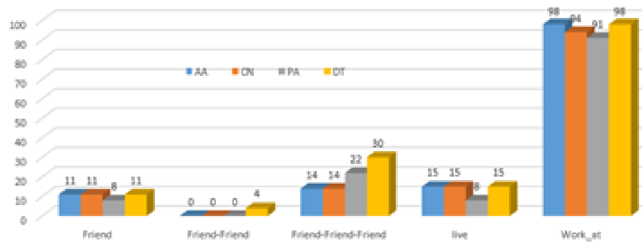

Figure 5. Similarity value matches according to relationships

V. CONCLUSION AND FUTURE WORK

Measuring trust among users on social networks has great importance for decision making friends. Relation prediction in social networks has attracted increased attention from policy makers in many business disciplines. Calculating trust between users is very useful for users to discover friends around or members of organizations, thereby helping that user and their friends to expand your network. friends, have many opportunities to exchange and cooperate. In actual society, there are many types of relationships between people such as friends, colleagues, relatives, classmates, partners, etc.

To calculate the trust between users in a multi-relationship network, we should take into account the importance of relationships. In this article, we consider four types of relationships and rank them in order of priority from high to low: your friend relationship, your friend's relationship, the same place of work and the same place of residence. Experimental results show that our proposed DT algorithm is superior to other similar algorithms and can achieve more accurate results on multi-relational networks.

In many real-world social networks, users may have attributes such as age, gender, occupation, income, habits, interests, etc., which are valuable for calculating the credibility of Surname. In our current study, we omit such useful information about user properties. Because, the actual collected data on Facebook of users often leave this information blank. However, the data of the users in the social network changes over time. Therefore, our future work is to learn efficient methods to integrate multiple user attributes for inclusion in the reliability algorithm.

REFERENCES

[1] Neo4j, "Neo4j Documentation", https://neo4j.com/docs/, accessed date: March 10th, 2022.

[2] https://neo4j.com/docs/graph-data-science/current/algorithm, accessed date March 10th, 2022.

[3] https://docs.redis.com/latest/stack/deprecated-features/graph/, accessed date March 10th, 2023.

[4] http://orientdb.org/, accessed date March 10th, 2023.

[5] Cypher Query Language, https://neo4j.com/developer/cypher/, accessed date March 10th, 2023.

[6] Iftikhar Ahmad, Muhammad Usman Akhtar, Salma Noor & Ambreen Shahnaz, *Missing Link Prediction using Common Neighbor and Centrality based Parameterized Algorithm*, Sci Rep 10, 364 (2020). https://doi.org/10.1038/s41598-019-57304-y

[7] Sourabh Dadapure, *Recommendation Engine using Adamic Adar Measure*, American Society for Engineering Education, 2022.

[8] Sheridan, P., Onodera, T., *A Preferential Attachment Paradox: How Preferential Attachment Combines with Growth to Produce Networks with Log-normal In-degree Distributions*, Sci Rep 8, 2811 (2018). https://doi.org/10.1038/s41598-018-21133-2.

[9] Pham Thi Thu Thuy, *Algorithm to determine the trustworthiness of friends on social networks based on Ontology*, Science Journal of Da Lat University Vol. 8, Issue 2, 2018 139–150.

[10] Graph API, https://developers.facebook.com/docs/graph-api/, March 10th, 2023.

[11] Ling Chen, Man Gao, Bin Li, Wei Liu, Bolun Chen, *Detect potential relations by link prediction in multi-relational social networks*, Decsup (2018), doi:10.1016/j.dss.2018.09.006.

[12] Kai Zhu, Meng Cao, Heng-yang Lu, *MALP: A More Effective Meta-Paths Based Link Prediction Method in Partially Aligned Heterogeneous Social Networks*, 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI).

[13] Ling Chen, Man Gao, Bin Li, Wei Liu, Bolun Chen, *Detect Potential Relations by Link Prediction in Multi-Relational Social Networks*, Decsup (2018), doi: 10.1016 / j.dss.2018.09.006.

[14] Pinghua Xu, Wenbin Hu, Jia Wu, Bo Du, *Link Prediction with Signed Latent Factors in Signed Social Networks*, Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA. ACM, New York, NY, USA, 9 pages.

[15] Virinchi Srinivas, Pabitra Mitra, *Link Prediction in Social Networks*, Springer, 2016.

[16] Jan Skrasek, *Social Network Recommendation using Graph Databases*, Brno, 2015.

[17] Gorka Sadowski, Philip Rathle, *Fraud detection: Discovering Connections using Graph Databases*, The World's Leading Graph Database, January 2015.

[18] DB Engines, *Knowledge Base of Relational and NoSQL Database Management Systems*, https://db-engines.com/, accessed date: March 20th, 2022.

Pham Thi Thu Thuy received the B.Eng. degree in Computer Technology from Hanoi University of Science and Technology, Vietnam, in 2001. In 2006, she received the Master degree, and in 2012 received the Ph.D. degree, both on Computer Engineering from Kyung Hee University, South of Korea. She is a main lecturer of the Nha Trang University, Vietnam. Her research interests include ontology matching, semantic measurement, graph database, ontology and Semantic Web.

Email: thuthuy@ntu.edu.vn

Nguyen Thi Thai Thanh received her bachelor's degree on information technology in 2011 from University of Information Technology, Ho Chi Minh City National University. She has got the master degree in Computer Technology at Nha Trang University. Currently she is working at the Training Department of Van Ninh Vocational School in Van Ninh district, Khanh Hoa province, Vietnam.

Email: thaithanhcn@gmail.com

Hwa Soo Kim received the B.Eng. degree in Electronics from the Korea University, Seoul Korea in 1981. In 1984, he received the Master degree in Computer Science from the Naval Postgraduate School, United States and his Ph.D. degree at Case Western Reserve University, United States in 1990. He was a professor of the Korea National Defense University and is a Special Appointment Professor at AJOU University, Korea. His research interests include cyber security, project management, AI technologies.

Email: ajhskim@ajou.ac.kr