Future Friend Recommendation System based on User Similarities in Large-Scale on Social Network

Md. Amirul Islam[1], Linta Islam[2], Md. Mahmudul Hasan[3], Partho Ghose[4],
Uzzal Kumar Acharjee[5], Md. Ashraf Kamal[6]
Department of Computer Science and Engineering, Jagannath University, Dhaka, Bangladesh[1,2,3,4,5]
Department of Computer Science and Engineering, World University of Bangladesh, Dhaka, Bangladesh[1,6]

***Abstract*—Friendship is one of the most important issues in online social networks (OSN). Researchers analyze the OSN to determine how people are connected to a network and how new connections are developed. Most of the existing methods cannot efficiently evaluate a friendship graphs internal connectivity and decline to render a proper recommendation. This paper presented three proposed algorithms that can apply in OSN to predict future friends recommendations for the users. Using network and profile similarity proposed approach can measure the similarity among the users. To predict the user similarity, we calculated an average weight that indicates the probability of two users being similar by considering every precise subset of some profile attributes such as age, profession, location, and interest rather than taking the only average of the superset profile attributes. The suggested algorithms perform a significant enhancement in prediction accuracy 97% and precision 96.566%. Furthermore, the proposed recommendation frameworks can handle any profile attribute's missing value by assuming the value based on friends' profile attributes.**

Keywords*—Social networks; recommendation framework; profile similarity; network similarity

I. Introduction

OSN is a platform for sharing information (such as opinions, news, views), communication, business over the internet and related with the connectivity of people. The popularity of OSN increases day by day in recent times, and OSN data are referred to as one of the most important sources of information over the internet [1]. It is a great medium to be connected with more similar and both known and unknown people to share their own opinions and do audio or video chat. The OSN provides the facility for spreading news over the internet, helps to smooth communication with individuals or the community quickly, and helps continued other internet-based activities (such as online shopping and blogging).

An OSN is interpreted as a graph $g = (v, e)$, where users are denoted as node v, and the relation between users is denoted as edge e. Online social networking encloses networking for business, pleasure, and all points in between users. Networks themselves have different objectives, and their online reproduction work in many ways. A social network allows people to share information with friends and familiarity, both old and new. Therefore, every user creates a personal network based on some user properties and wants to broaden their network to create a new friendship using profile and network similarity. A user in a network creates a new friendship link with others for communication or sharing views, opinions or other information after creating an account. Mainly two points are involved in creating a new relationship in the social network by the theory of Homophily [2]. The first point is that users try to establish relationships with other users based on who is closer to them on the social graph. And the second point is users form relationships with users who are comparable to them and have particular properties like occupation, age, religion, hobby, gender, etc.

In social networks, how users are connected can be known by analyzing social networks. We can find the hidden patterns of the network and which path is best for spreading news, advertisements and political opinions. Analyzing online social networks helps us how to impact social media on human behaviours and how social media use in a convenient way [3]. Facebook[1], Twitter[2], VKontakte[3], Flickr[4], YouTube[5] are the most popular OSN attracted people by their impact on the internet as an excellent media for sharing news, opinions, interest, pictures, videos and a great communication medium.

Network similarity indicates the similarity among different networks rather than other nodes in a social network. Each user in a social network has their sub-network with friends and friendship links, and with time, users want to broaden their network for information sharing by including new friends. People want to establish new friendships with others who are closer to each other in OSN. Two graphs are used to compute the network similarity in the social graph. One is a friendship graph, and another one is the mutual friends' graph. The profile contents are unstructured keywords such as education, profession, gender, age, interest, and one or more of these are used for finding similarities between profiles. The string matching method is used to calculate the profile similarity among the profile attributes. In the paper [4], authors calculated the profile similarity of Facebook by handled only the individual profile value (Interest) of the users. On the other hand, authors in the paper [1] calculated the profile similarity of Facebook by considered user occupation, education, and gender.

Today OSN has become a large field of online activities, and it continues broadening day by day. So, it becomes a complex topic for a user to find a similar account from an extensive network. Information interchange is a common phenomenon in social networks. Those activities bring an

[1] https://www.facebook.com/
[2] https://twitter.com/
[3] https://vk.com/
[4] https://www.flickr.com/
[5] https://www.youtube.com/

excess of messages that make users confused about what they want. OSN recommendation system [5] is a framework that recommends the user to others by analyzing their available social information. Information of users in online social networks are largely available, mining profile information which can be able to predict user personality, that is an essential issue for finding user preference for recommending products, songs, online game mining social data [6]. In social networks, users are introduced to each other in several ways. Friend matching is a technique that can help to find friends on social media. Users connect using similar profile information such as similar educational background, similar home town or same interest.

In most cases, a recommendation system fails to recommend a similar use on the web because of missing profile content [7]. For constructing a recommendation system, it is essential to confirm that the user profile contents are available. Some profiles in OSN cannot provide us with all attributes information; in this case, the recommendation system fails to recommend appropriately. In order to overcome this problem, we need a technique that can infer profile missing value. Therefore the primary aim of this paper is to mine an extensive group of social data and discover the more related people or users for the recommendation. Our suggested approach computes the similarity by utilizing various similarity measures among all probable peoples and recommends them if they are not friends still now.

Following are the contributions of this paper:

- Analysis of online social networks to find user similarity between two users by combining missing profile items, network similarity, and the weight of each attribute set for profile similarity.
- To propose prediction of profile disappeared value (Algorithm 1) to handle missing profile values.
- To construct a better future friend recommendation system for users, we presented two modified profile weight calculations methods named Feature Weight Computation System (FWCS) and Friend Matching System (FMS).

The remaining part of this paper is outlined as follows: In Section II, literature review described. Our proposed methods and algorithm with calculation of user similarities for friend recommendation system are discussed in Section III. Experimental results and discussion is shown in Section IV and in Section V, briefly concludes our research effort with future research directions.

II. Literature Review

Focusing on profoundly significant work, we audit some current related work and afterwards sum up standard techniques for recommendation framework. This section briefly presents a few studies handle in recent years by different strategies.

Friendship is a fundamental relationship in social networks and suggests friends are practical activities to overcome this, [8] proposed a friendship recommendation solution by profile matching. This work assigns different weights in different items and developed a mining model to discover different factors actual degree of influence by measuring the profile attributes using some similarity measures. The proposed framework yielded an accuracy of 95%. In recent years, a similarity measure between nodes is defined based on the features of their neighbourhood information from many users in an extensive network. In the paper [9], suggested parametric system for neighborhood-based similarity is applied to calculate several similarities and calculative costs among neighborhood nodes. A unique multi-feature SVM based friend recommendation model (MF-SVM) is introduced by Xin et al. [10]. This proposed model is a binary classification problem. It can handle the sparse situation of user location and user-user formation in the location-based social network. The authors extracted three features using their proposed model but did not consider or handle missing values, network similarities, and weight calculation. To evaluate the MF-SVM model, two real-world data sets, Foursquare and Gowalla are chosen. The model achieved an accuracy of 90%.

In 2020, Qader et al. [11] suggested a Dual-Stage FR model to recommends users to other users based on user interests. The model applies the double stage technique on unlabeled information of 1241 users collected from OSN users via the online survey. The authors mainly combined user-based collaborative filtering (UBCF) and graph-based FR in their proposed model. However, the drawback of this technique is that the computational cost linearly increases with the user. The accuracy of the model was 86%. In 2020, Soni et al. [12] presented a novel FR framework based on their similar choices, activities, preference and locations. The authors replaced k-means clustering with hierarchical clustering in their proposed model and principal component analysis (PCA) techniques applied to the dataset for dimensionality reduction. However, the limitation of this model is the cost of PCA calculation when the matrices become high. The model achieved an accuracy of 89.47%.

Kumar et al. [13] introduced a graph-based FRS utilizing two CF systems: the number of mutual users and the influence factor. Then, it assigned a score number to every conceivable friend to track down the higher closeness between clients dependent on the highest score number. The datasets utilized are Stanford SNAP, which individually consists of 4039 and 81,306 clients from Facebook and Twitter. The model achieved an accuracy of 97.2%. In the research paper [14], a new framework is proposed called multi-step resource allocation (MSRA) to predict the implicit relationships. The authors are mainly combined three sources of information: a user-item matrix, explicit and implicit associations. To evaluated the proposed method, two real datasets are used (Last.Fm and Ciao). The proposed MSRA model achieved an accuracy of 95.80%. To predict the future friends in the social networks, Shabaz et al. [15] proposed a new approach called Shabaz–Urvashi Link Prediction (SULP). This new technique can solve the problem of linking isolated or missing nodes in social networks and connect the nodes in a network faster than any other link prediction algorithm that exists. For this reason, this novel approach can reduce the connection time and resources involved in it. The proposed SULP model achieved a precision of 76%, recall of 82% and TRP of 88%. In 2021, Berkani et al. [16] proposed a unique recommendation framework for users in social networks. This method mainly based on semantic and social-based classification of the user profiles. The authors have used two classifications techniques: the K-means algorithm and K-Nearest Neighbours algorithm to optimize the performances of the recommendations systems. This proposed model used two datasets, one is the Yelp datasets, and another one is the Rich Epinions datasets. The proposed method achieved an accuracy of 95%.

Apart from this, researchers continuously contribute to developing an efficient system to recommend friends to the users. We have taken some recent papers, and their contribution in a different part of similarity measure is shown in Table I.

It is shown that different parts of similarity measurement are not fulfilled. For this reason, existing techniques could not measure user similarity efficiently, and the recommendation system could not recommend properly. In this paper, we proposed an efficient technique for measuring user similarity between two users, combining all parts (inferring missing profile item, network similarity, the weight of each attribute set for profile similarity) of similarity measurement and an efficient recommendation system. The existing method measuring network similarity uses mutual friendship, and target user friendship graph edges only. In a new friendship formation, two users have the same influence. In our network similarity, method friendship graph edges two of them are used. This work used only observed frequency measure in profile similarity and set the same weight to each profile attribute. Utilizing weight computation of every profile attribute's performance by considering only profile similarity, authors recommended future friends for the users in their paper [8]. But our proposed framework has diverged from other research contributions because firstly, we computed the weight for every set of profile attributes and then merged it with the network similarity. For creating future friendships among the users', the profile attributes set contributed a vital influence. In our proposed method combine feature computerization systems based on the supervised learning strategy.

III. Research Methodology

This section represents the main portion of the paper: the proposed architecture design and development of the proposed algorithms.

A. Proposed Architecture Design

Friendship and profile information from the online social network is used in our suggested model to calculate the similarity of several users who do not belong to the friendship graph. There are four phases in our proposed model: The first phase is used to extract the user's features based on the user profile and handle the user's profile attributes if there are any missing data by assuming the value. Data mining technology is used in the second phase. In the third phase, the friend matcher method recommends the future friend for the user by predicting the user's similarity. And in the final phase, the feature automation technique (Supervised Learning-Based) is used to formation the friendship by identifying the most prominent attributes. In Fig. 1 demonstrated the proposed model architecture. All parts of this model are described in detail below.

1) User Profile: Each user in a social network has two types of information: profile information and friendship information. The profile holds some user personal information such as name, home location, date of birth, gender, profession, interests, educational information, etc. This paper considers only four profile attributes and friendship information to unlock the critical fact of making a new friendship. A profile of four attributes: home location, profession, date of birth and interest.

2) Features Extraction: In this step, our proposed model can extract the information (home location, profession, date of birth and interest) from a social network by using some API.

3) Handling Missing Value: A social network consists of a large number of user profiles. Every user profile has personal information such as name, email, location, hometown, date of birth, personal interests, profession, gender etc. Some attributes have multiple values (e.g. interests = [programming, football, reading, gardening]). It is highly possible that some individuals do not possess all types of attributes, and some specific attribute's values may be missing in the profile. Thus, while comparing two user profiles, it is great to have all the information to measure similarity or dissimilarity. If one or more fields of a profile are missed, comparison cannot be performed, and hence it does not allow similarity among an individual's profile. For this reason, inferring missing profile items is an essential part of similarity measurement.

Handling missing values of a profile is just like making a data preprocessing. In [18], the authors proposed inferring personal items of a profile. The approach of calculating missing items is usually made by taking into account all of the information of friends or by searching a user's group membership. To disclose political views or sexual relationships can be obtained by accumulating all friends' profile information or group membership. In most cases, for security purposes, the social network does not allow this access, and it is not possible to extract all information of friends or group membership. Moreover, information cannot be extracted because some social users hide their sensitive information. As a result, some information about any user or his friends cannot be extracted. It is possible to retrieve user profile information with the help of social network $APIs$.

Considering all those limitations, we propose a method to overcome the problem of missing profile items. To infer a user profile information by using all of his/her friends profile information, we find the rank of each attribute's value of this missing profile item. The highest rank is considered as a value of this missing profile attribute. We have used a modified page rank algorithm to find out the highest rank of missing profile items. We calculate the vote of each friend according to his profile attribute value which is missed by a user u. The modified page rank formula is defined in Equation 1.

$$R(D_i) = P(D_i) + \frac{\sum_{j=1}^{n} \frac{P(D_i)}{1+\neg P(D_i^j)}}{n} \tag{1}$$

Here, $R(D_i)$ $P(D_i)$ refers rank and probability of a attribute respectively, $\neg P(D_i^j)$ represent probability of the other attributes except the attribute D_i of j^{th} user. Here, we add 1 with $\neg P(D_i^j)$ because of $\neg P(D_i^j)$ may zero when one user not connected with different profile attribute value user.

TABLE I. CONTRIBUTION OF DIFFERENT PAPER IN DIFFERENT PORTION

Existing Paper	Missing Item	Network Similarity	Determining Weight	Feature Automation
Akcora et. al. [1]	✓	✓	✗	✗
Msazhari et. al. [8]	✗	✗	✓	✗
Xin et al. [10]	✓	✗	✗	✓
Qader et al. [11]	✓	✓	✗	✓
Soni et al. [12]	✗	✓	✗	✗
Kumar et al. [13]	✗	✓	✗	✓
Al-Sabaawi et al. [14]	✓	✗	✗	✓
Shabaz et al. [15]	✗	✗	✓	✓
Berkani et. al. [16]	✗	✓	✓	✓
Razis et. al. [17]	✗	✗	✗	✓

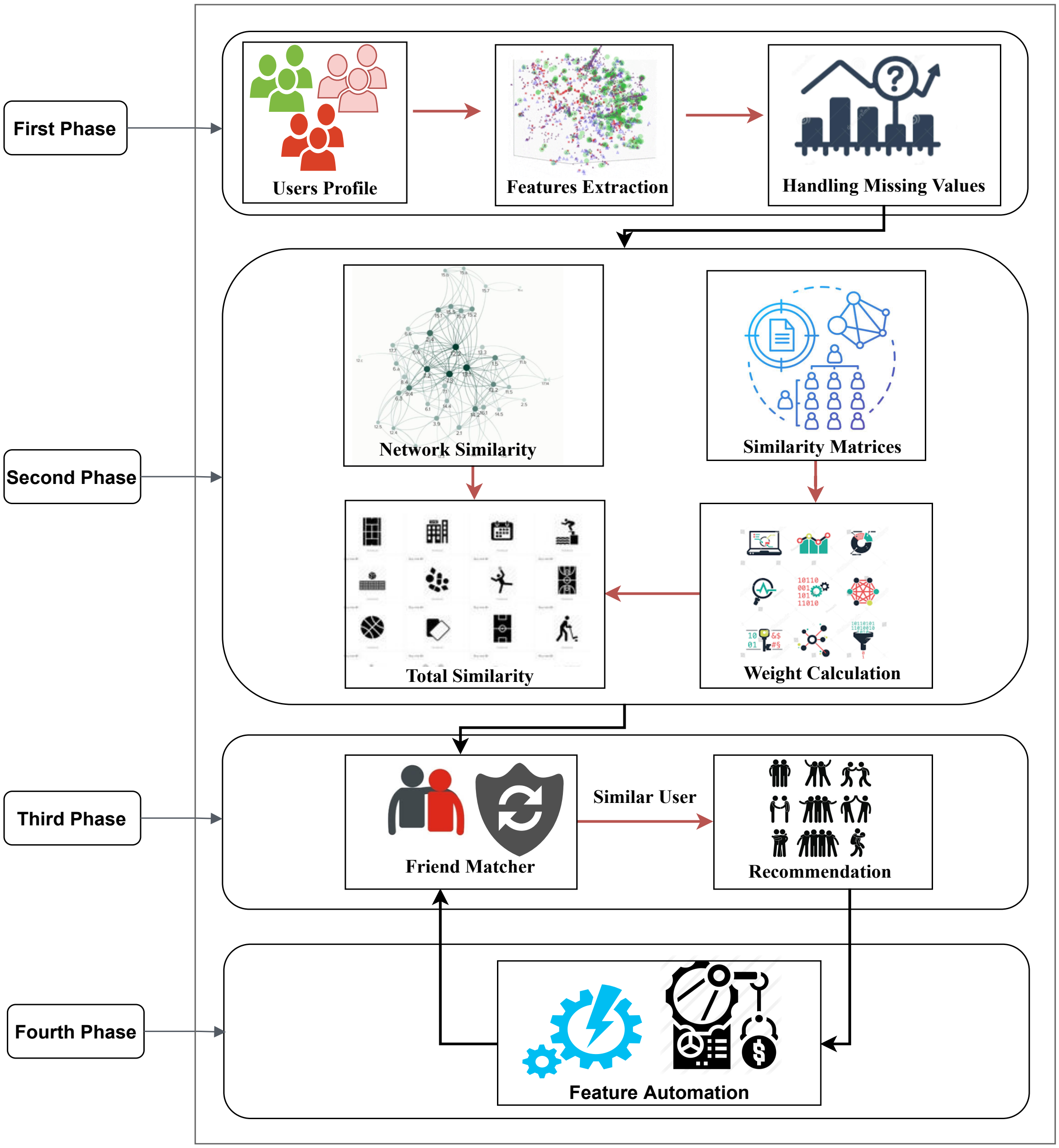


Fig. 1. Proposed Architecture for Recommendation System.

4) User Similarity: To measure similarity between two users u and v, where $(u, v) \in G.V$ in a considered network $G(E, V)$, where E represents friendship link and V represent user, we proposed a modified NS for network similarity and modified weight calculation algorithm for profile similarity, we use network and profile similarity for user recommendation.

We describe user similarity part in three phases: network similarity, profile similarity, friend matching.

A.1 Network Similarity: In a social network, people like to create new friendships with those closest to each other in the social graph. They create new edge and contact with those friends. Here we compute network similarity between two users u and v who are not friends and the closeness between u and v based on their information. Most effortlessly, network similarity can be computed by only using the number of mutual friends of u and v. In this approach, the only used node of friendship graph as a result in some case important information can be loosed, and the performance of network similarity is not good. In [1] used the edge of the social network mutual friendship graph for calculating network similarity between two node distances of two users. In mutual friends, there is no friendship relation between u and v and the mutual friends represent a complete graph called mutual friends graph as a subgraph of our target graph $G = (E, N)$, where N is the set of nodes and represent the social network users, the set of E represent the relationship among the social network user. In the previous measurement, they used node of mutual friendship graph, it cannot provide the relationship among mutual friends, but in mutual friend's edge set among the considered nods, includes all relationship information. Mutual friends graph formally defines as below:

Definition 1. *(Friendship Graph).* In a social network G, u is a node $u \in G.V$, friendship graph of u denoted as $FG(u)$ is a sub-graph of G where, $FG(u).N = \{u\} \cup \{n\}$, where $\forall n \in G.N, n \neq u, \forall e \in FG(u).E, e \in < u, n >$ and $FG(u,v).E = \{< x, x' >, \forall x \in FG(u,v).N, \forall x' \in FG(u,v)$.

Edges represent more information about friendship among considered users in a social network rather than nodes. Edge count provides how strongly the users tie each other. The existing work computes the similarity between u and v; they only compare the mutual friend's graph and the target user $u's$ friendship graph edge count. A friendship graph of u consists of $u's$ entire friend in a graph and all edges among the node. The formal definition of friendship graph is as follows:

Definition 2. *(Mutual Friends Graph).* In a social network G, u and v are two nodes $u, v \in G$, mutual friends graph of u and v, denoted as $MFG(u,v)$ is a sub-graph of G where, $MFG(u,v).N = \{u,v\} \cup \{FG(u).N \cup FG(v).N\}$ and $MFG(u,v).E = \{< u, x > \in G.E \cup < v, x > \in G.E\}, where x \in MFG(u,v).N, x \neq u, x \neq v$.

We use modified network similarity (NS) [1], the network similarity between two users in a social network graph can be computed by the ratio of the number of edges in the mutual friends' graph of u and v and the sum of the number of edge in the friendship graph of u and v.

Definition 3. *(Network Similarity).* Network similarity between two users u and v is defined as:

$$NS(u,v) = \frac{log(|MFG(u,v).E|)}{log(|FG(u).E + FG(v).E|)} \quad (2)$$

Where, $|MFG(u,v).E|$ represent the number of edges in $MFG(u,v)$ and $|FG(u).E + FG(v).E|$ represent the total number of edges in $FG(u)$ and $FG(v)$.

The existing method used only the ratio of $MFG(u,v)$ and $FG(u)$. In the proposed method, we used both u and $v's$ friendship graph edges count. Because of finding out the influence both u and v on mutual friends graph. If u and v have no mutual friends, mutual friends graph remain two nodes but no edge, i.e. $MFG(u,v).V = u, v$ and $MFG(u,v).E = 0$. In this case, the value of network similarity is zero.

A.2 Profile Similarity: Every profile contains some unstructured keywords, and these are associated with the user's details. The profile similarity between u and v can be calculated by measuring the similarity between the same profile items of two profiles. In this paper, we have taken four profile attributes: age, home location, profession, and interest to measure the similarity between profiles. Similarity between two users u and v depends on the similarity value between items $u_{interest}$ and $v_{interest}$, u_{age} and v_{age}, $u_{profession}$ and $v_{profession}$, $u_{location}$ and $v_{location}$. Profile items are heterogeneous, so it is harder to measure the similarity of different items by applying only one similarity measurement formula. However, there have some suitable similarity measures for every specific type of attribute. In this paper, three similarity measures are used to calculate the similarity between the items. Damerau–Levenshtein distance [19], Levenshtein distance [20] and Manhattan Distance [21], which are used to determine similarity of location, profession, age and interest.

A.3 Weight Calculation: Each profile consists of personal information such as age, interest, profession, location etc., and some standard measuring techniques are used to calculate the similarity between profiles attribute. In most existing techniques, only the information of two profiles of the recommended persons is used. Very few research works consider the influential factors of the existing social network to recommend new friends. It is more practical to measure the influence of profile attributes of the existing friends to predict new friend matching. In this case, the authors of [8] use a single set of some attributes to find out the influential factors of the existing network. Nevertheless, we consider both single and multiset attributes to calculate the influential factor of the current network. The multiset of attributes is often responsible for forming new friendship links in real life. For instance, two persons of the same age and interest are more likely to be friends than two persons with only the same age or only the same interest. In that sense, we introduce the concept of a multiset of attribute's comparison to recommend a new friendship link. Only the weight of each attribute is not sufficient for proper friend matching. For efficient fiend matching techniques, we need to compute all sets of attributes. Multiset attribute similarity can be calculated using the following Equation 3.

$$SV_{a_1,a_2....a_n} = SV_{\{a_1\}} \times SV_{\{a_2\}} \times ..... \times SV_{\{a_n\}} \quad (3)$$

5) Friend Matching Method: We have used a friend matching method (FMM) that calculates the similarity among two users u and v in two steps. Firstly, we calculate network similarity among two users (u & v) by applying Equation 2 and profile similarity using the following Equation 4. Secondly, it compares both profile and network similarity values with a threshold (TH). If one similarity value is greater than TH, it provides similarity between u and v otherwise dissimilarity. If both network and profile similarities are greater than the TH value, it provides a strong similarity between them. The probability of new link formation increases with the similarity value.

$$PS = \sum_{i=1}^{n} W_i * SV_i \tag{4}$$

In this equation n is the total attributes, W_i = Weight of i^{th} attribute set (e.g. $W_1 = W_{\{age\}}$,$W_2 = W_{\{location\}}$, $W_4 = W_{\{age,location\}}$). SV_i = i^{th} attribute similarity between u and v (e.g. $SV_1 = ageSimilarity(u_{\{age\}}, u_{\{age\}})$ $SV_2 = ageSimilarity(u_{\{location\}}, u_{\{location\}})$). All the similarities among the user will be calculated by using this equation.

In new friendship formation, users in an OSN observe profile attributes or mutual friends or both profile and mutual friends. If there is a similarity in profile attributes, it provides a probability of creating a new friendship link. Besides, a sufficient number of mutual friends provides a possibility to create new friendship links. Moreover, network similarity significantly impacts friendship formation when a user does not update his profile information.

Profile similarity (PS) and network similarity (NS) are independent. Both have an individual influence on user similarity. We use conditional probability To calculate the effect of both NS and PS on user similarity.

Here, total similarity value, $TSV = NS + PS$

$$P(PS|TSV) = \frac{P(PS)}{P(TSV)} \tag{5}$$

$$P(NS|TSV) = \frac{P(NS)}{P(TSV)} \tag{6}$$

$$\neg P(PS|TSV) = 1 - \frac{P(PS)}{P(TSV)} \tag{7}$$

$$\neg P(NS|TSV) = 1 - \frac{P(NS)}{P(TSV)} \tag{8}$$

Here, $P(PS|TSV)$ refers to the conditional probability of how much PS affects TSV, and $\neg P(PS|TSV)$ provides PS not effect to TSV. Calculating user similarity, we rank for each user who does not have a friendship link and a more significant user similarity value than a threshold value. The top of the rank table has the highest similarity. From the rank, table select top k user and recommends to u.

6) Feature Automation: In the case of recommendation of users, there will create a new friendship link. However, in all cases, all recommended users will not be able to create friendship links. For this case, a feature automation technique is introduced here to extract newly friendship link created user information. Firstly, this technique analyzes collected information and calculates each pair of user profile similarity using our three considered similarity measures to calculate attribute set similarity. When several profile pair similarities are calculated, it measures each attribute set's weight using Equation (3) to (8). The effective weight for each attribute set tries to exact the hidden fact that primarily influences creating new friendship links. It compares this calculated weight with the previous weight and which set is more affected and less. According to this decision, it updates attributes sets weight. Friend Matching Method uses the weight information and measures user similarity, and users will be recommended appropriately, and the outcome is better than previous.

B. Proposed Algorithms

In our research worked we approached three algorithms for the recommendation system. The first algorithm computed the missing profile values of the OSN users. Another two algorithms are moderated algorithms of [8]. Algorithm 2 is used to compute the weight of all conceivable sets of contemplated profile attributes. Algorithm 3 is used to estimate the network and profile similarity among the users, then recommended the future friend for the user.

1) Disappeared Value Estimation Technique: Our proposed Algorithm 1 can compute the missing items of the users' profiles. It is mainly a data preprocessing method. To predict the missing profile item of any user, this algorithm firstly discovers the missing profile items. In the algorithm, line number 5 indicates that the user's missing profile item is calculated and a probability computation function $CalculateProbability(p_{ij}, P_u R_i)$ is called in line 6. This function is mainly all friends missing attributes and compute the possibility of items and finally, calculate the largest estimation of item's value for that disappeared item. In our previous work [22], we described more about this algorithm.

Algorithm 1: Prediction of Profile Disappeared Value

```
   Input  : $P_u = \{p_1, p_2, p_3, .....p_n\}$      //users profile
   Output: Predicting disappeared values of every profile
1  $D =< Location, Interest, Age, Gender >$
2  foreach $p_i \in P_u$ do
3      $P_u R_i$ =Extracting friends of $p_i$
4      if $P_i[D_k] = NULL$ then
5          foreach $j \in P_u R_i$ do
6              if $p_{ij} = NULL$ then
7                  $pF_i = FriendsofP_i$
8                  foreach $j \in P_u R_i$ do
9                      $X_c$ = Counting $D_k$ not equal
                         $P_u F_i^j inMFG(P_u F_i^j, P_i)$
10                     $Y_v$ = Vector of pair number
11                     Discover Rank of $X_c$      //using Equation (1)
12                     $Y_v$.Push(Rank)
13                 end
14             end
15         end
16         Extraction Max($Y_v$)
17         Discover $P_u F_i[D_k]$ for maximum value
18     end
19 end
```

2) Feature Weight Computation System: Algorithm 2 was utilized to compute the weight of every set of attributes. In algorithm 2, lines 5 to 7 indicate that Manhattan Distance, Levenshtein Distance and Demaru Levenshtein Distance are used to compute the similarity of every user profile attributes with his all friends profile. To calculate the similarity of the profile attributes (profession, interest, & location), Levenshtein Distance and Demaru Levenshtein Distance are used. And calculated the 'age' similarity of the profile with the help of Manhattan Distance. In line 6 was used for calculating user attributes similarity and took the better value of similarity. In lines, 8 to 19 are used to compute the similarity for every user.

Algorithm 2: Feature Weight Computation System (FWCS)

```
    Input  : $P = \{p_1, p_2, p_3, .....p_n\}$
    Output: Weight of each feature
 1  $p_i =< profession, age, interest, location >$
 2  $i \leftarrow 1$
 3  foreach $p_i \in P$ do
 4      $PR_i$ =Extract friends of $p_i$
 5      foreach $j \in PR_i$ do
 6          $DLD_{location} += DemaruLevenshteinDistance(p^i_{location}, PR^{ij}_{location})$
            $DLD_{interest} += DemaruLevenshteinDistance(p^i_{interest}, PR^{ij}_{interest})$
            $DL_{profession} += LevenshteinDistance(p^i_{profession}, PR^{ij}_{profession})$
            $LD_{location} += LevenshteinDistance(p^i_{location}, PR^{ij}_{location})$
            $LD_{interest} += LevenshteinDistance(p^i_{interest}, PR^{ij}_{interest})$
            $DM_{age} += ManhattanDistance(p^i_{age}, PR^{ij}_{age})$
            $DL_{profession} += LevenshteinDistance(p^i_{profession}, PR^{ij}_{profession})$
 7      end
 8      if $DLD_{location} > LD_{location}$ then
 9          $D^i_{location} = \frac{DLD_{location}}{|PR_i|}$
10      else
11          $D^i_{location} = \frac{LD_{location}}{|PR_i|}$
12      end
13      if $DLD_{interest} > LD_{interest}$ then
14          $D^i_{interest} = \frac{DLD_{interest}}{|PR_i|}$
15      else
16          $D^i_{location} = \frac{LD_{location}}{|PR_i|}$
17      end
18      $D^i_{age} = \frac{DM_{age}}{|PR_i|}$
19      $D^i_{profession} = \frac{DL_{profession}}{|PR_i|}$
20  end
21  $W_{\{s\}} = \frac{\sum D^i_{\{s\}}}{|P|}$   /* Calculated weight for each attribute set, here, $s$
      represent attribute set*/
```

We consider all probable profile attributes sets to calculate the weight in our suggested algorithm, but in the paper [8], authors are only considered profile attributes to calculate the weight. On the friend matching, this weight of profile attributes creates various impacts. The presence of attributes is called an attribute set. The multiplication process is used to calculate the set value of attributes. Example: multiplication of $similarity_{\{profession\}}$ and $similarity_{\{gender\}}$ is produce $similarity_{\{profession,gender\}}$. As we know that, if we multiply two positive numbers, the result will be too smaller if both numbers are less than one. So, the attribute similarity set value becomes small if any attribute similarity is small in the set, and also, the profile attribute similarity value will be too smaller. The average value of all set similarities values is called profile similarity value. In that circumstance, dissimilar profile values will be nearest to 0, and similar profiles will be nearest to 1. By this technique, the proposed algorithm can easily discover the dissimilar and similar profiles.

3) Friend Matching System: We calculated user similarity among pairs of users (u & v) using Algorithm 3 after measuring every attribute set's weight.

In this proposed algorithm, lines 3 to 6 indicates that every profile attributes similarity is calculated. LocationSimilarity mentions the most suitable similarity value among Levenshtein and Demaru-Levenshtein similarity. NS and PS are calculated in lines 7 and 8. It generates a rank of similarity based on avg. of PS and NS; if the TH (threshold) is lowest than both NS and PS or one of them. From every user rank table, the topmost k users are recommended.

Algorithm 3: Friend Matching System (FMS)

```
    Input  : $P = \{p_1, p_2, p_3, .....p_n\}$   //from Algorithm 2
    Output: Rank of the similar user
 1  $p_i =< F^i_{age}, F^i_{gender}, F^i_{location}, F^i_{interest} >$
 2  foreach $i \notin P$ and $j \notin PR_i$ do
 3      $SV_{age} = ManhattanDistance(F^i_{age}, F^j_{age})$   //SV refers
          similarity value
 4      $SV_{profession} = LevenshteinDistance(F^i_{profession}, F^j_{profession})$
 5      $SV_{location} = LocationSimilarity(F^i_{location}, F^j_{location})$
 6      $SV_{interest} = InterestSimilarity(F^i_{interest}, F^j_{interest})$
 7      $NS = NetworkSimilarity(p_i, p_j)$   // using Equation (2)
 8      $SV\{s\} \leftarrow \{profession, interest, location, age\}$   // using
          Equation (3)
 9      $PS = \sum W_{\{s\}} * SV_{\{s\}}$   // using Equation (4)
10      $TSV = NS + PS$   //Total similarity value
11      $P(TSV|PS, NS) = \left[\frac{P(PS|TSV)*P(NS|TSV)*P(TSV)}{P(PS|TSV)*P(NS|TSV)*P(TSV) - \neg P(PS|TSV)*\neg P(NS|TSV)*\neg P(TSV)}\right]$   /* using
          Equation (5) to (8) */
12      if $P(TSV|PS, NS) > TH$ then
13          $Profile^i_j = P(TSV|PS, NS)$   //TH is the threshold value
14      end
15  end
16  $Rank_i \leftarrow sort(Profile_i)$ /* Users recommended based on rank of
      similarity */
```

IV. Experimental Results and Discussion

Here, we represent our experimental results and discuss the performance evaluation of our proposed algorithms. We used a real OSN dataset to evaluate the performance and examine the experimental results from different angles. Proposed algorithms are developed in the C++ language.

A. Dataset Collection and Description

In our experiment, we used the Facebook dataset and collected it from [23]. In this dataset, we discovered an accurate OSN undirected friendship graph. Four thousand (4K) different users and eighty-eight thousand (88K) edges are available in our dataset. Friendship is defined through searching for an edge among the pair of users. Many to many relationships for every user has been calculated on the whole dataset. Moreover, a vast number of similarity values are created from the dataset. We can compare it to an identical matrix where the matrix's upper and lower parts have the equivalent value. So, we calculated only one part from these two parts. As a result, complexity is decreased. Our experimental results on this dataset illustrated in the Table II.

B. Factor Coefficient

It is necessary to understand which items are more important for users, so considering the characteristics of similarity among all users of friends and the average similarity of each set of features has calculated from our considered dataset shown Table III with their coefficient. Each feature's coefficient is essential for friend matching. The Friend Matching Method

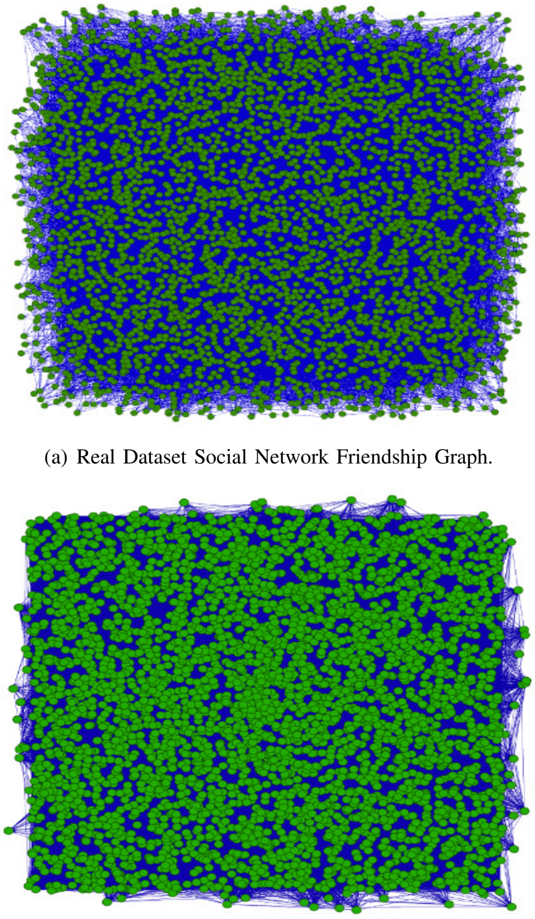
TABLE II. Statistics of Social Network

Hyper Parameter	Weight
Nodes	4039
Edges	88235
Nodes in largest WCC	4039 (1.00)
Edges in largest WCC	88235 (1.00)
Number of triangles	1612010
Fraction of closed triangles	0.2647

calculates the similarity between two users multiplying with this corresponding item set weight in friend matching.

C. Result and Analysis

Consequently, more than 55% of new similar edges have been established by using our proposed method, which is shown in Table IV. The experimental result is also inspiring when we applied it to the custom dataset. In Fig. 2(a), we show the actual social network friendship graph, and in Fig. 2(b) we additionally showed the resulting graph obtained after processing the dataset by our recommended method.



(a) Real Dataset Social Network Friendship Graph.

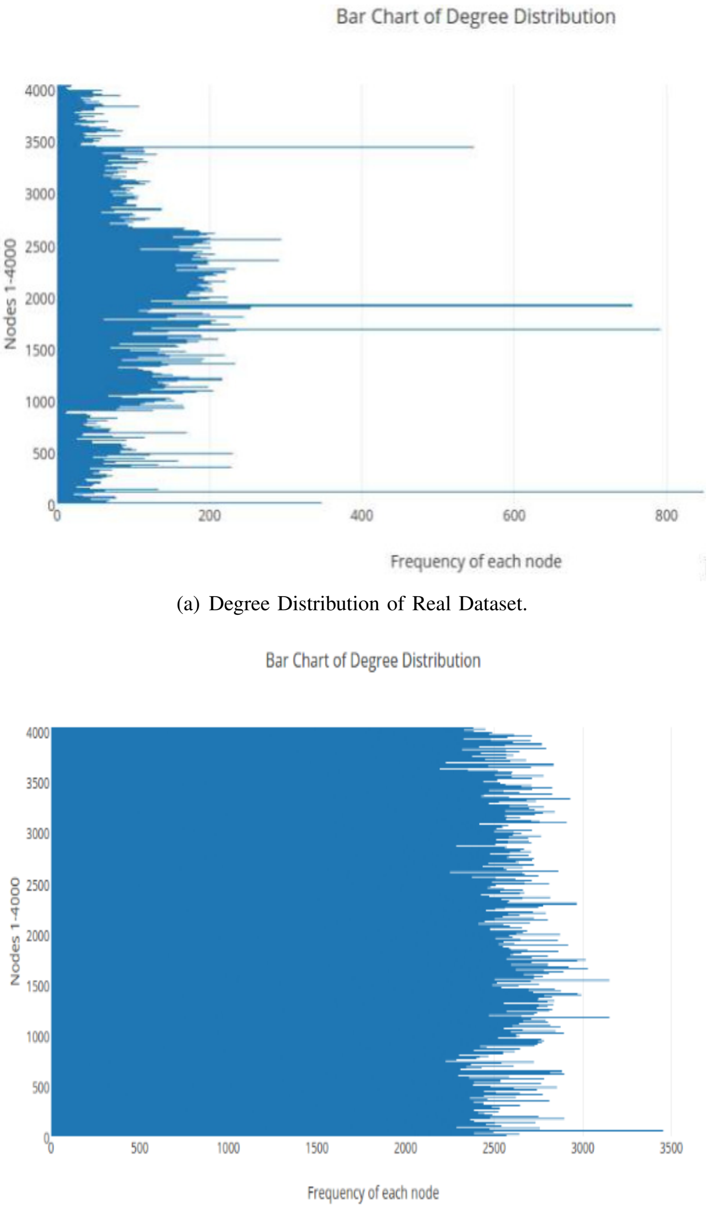
(b) Social Network Friendship Graph after Applying Proposed Formula.

Fig. 2. Friendship Graph.

D. Comparison of Degree

When we compare two distinct graphs, it is necessary to consider degree comparison. The average degree in the two undirected graphs is shown in Table V. on an average, only 43 edges are discovered before the network processing but using our method, 1953 edges are discovered after the processing.

For every complex network showing degree, the comparison is very fundamental and must be given case. To explain more, we have included the histogram of the frequency distribution of each node to describe the degree. From that evidence, we can be able to compare more efficiently. The Fig. 3(a) show real case of dataset and Fig. 3(b) shows the processed case of our dataset.



(a) Degree Distribution of Real Dataset.

(b) Degree Distribution of Processed Dataset.

Fig. 3. Degree Distribution.

From the above evidence, we can normalize the idea that

TABLE III. FACTORS COEFFICIENTS

Factor	{l}	{i}	{a}	{p}	{l,i}	{l,a}	{l,p}	{i,a }	{i,p}	{a,p }	{l,i,a}	{l,i,p}	{i,a,p}	{l,a,p}	{l,i,a,p }
Weight	0.791	0.440	0.741	0.588	0.351	0.588	0.471	0.338	0.286	0.457	0.271	0.231	0.191	0.345	0.152
Coefficient	13%	7%	12%	10%	6%	9%	8%	5%	4%	7%	4%	4%	3%	6%	2%

TABLE IV. STATISTICS OF SOCIAL NETWORK EDGE

	Before	**After**	**Increasing %**
Number of Edges	88,235	4,809,284	55.50

TABLE V. COMPARISON OF DEGREE

	Before	**After**
Networks Type **Degree(Avg.)**	Undirected 43.691	Undirected 1953.98

the small network number of groups can be suggested before processing. After processing, each group may have a higher number of profiles that will give more accurate friendship results or, in other words, a more efficient friendship network will produce. According to the presented tables and figures in the previous section, we can say that the proposed method has significantly impacted the friend matching system for this dataset.

As a further study of this experimental result, firstly, we divided it into three classes (A, B, C) based on the range of similarities for the friendships set and representing the ranges 0.0–0.49, 0.50–0.70, 0.71–1.00, respectively. Recommended friendships between the A, B, C classes are shown in Fig. 4. According to the calculation of similarities proposed recommended system recommended 39% future friends in class A, 50% in class B, and 11% in class C for the user. So, it's proof that the proposed approach effectively recommended more new relationships for the user. Furthermore, it delivers satisfying independence for other recommended friendships, which are not in the actual friendship graph.

E. Calculation of Accuracy

To estimate the accuracy, we used the probabilistic technique. Firstly, we randomly took few edges whose similarities measured and friends in an existing social network. Finally, from the existing network, we erased those edges. In the first scenario, 0.7157017 is the average similarity value, and 0.67801 is the average similarity value after erased the edges. So, after calculation, we see that our proposed algorithm accuracy is 97%. That indicated 97% users in real friendships graph and our proposed recommendation system can be capable of recommending the future friends for the users and 3% of real friendship which our proposed system has not recommended. Calculation of error rate is shown in Table VI.

F. Clustering Coefficient Comparison

The clustering coefficient is a prevalent measure for such real networks, especially for the social network, which shows how tightly bond groups exist in the network. Here in our approach, we use a small dataset to examine the clustering coefficient. The result is also promising. Average clustering coefficient comparison is shown in Table VII. For the local version of the clustering coefficient where it is calculated for each node and the values are shown by the following Fig. 5(a) where shows the scenario of before processing and the Fig. 5(b) shows the scenario of after processing and adding more edges.

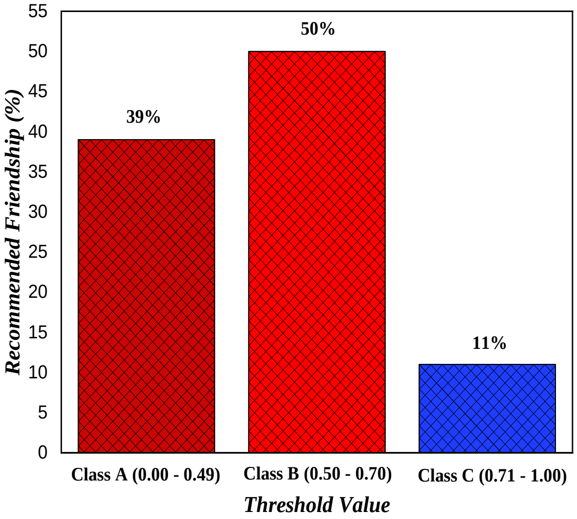


Fig. 4. User Friendships Distribution between the Three Classes.

TABLE VI. CALCULATION OF ERROR

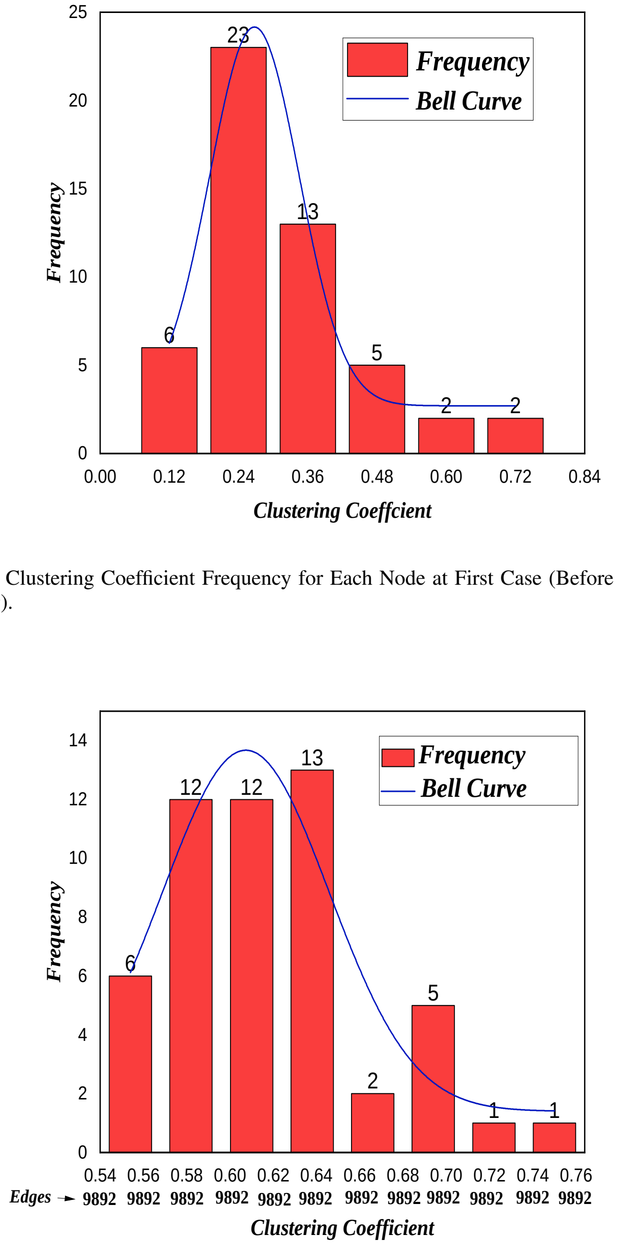
Similarity Value	**Before**	**After**	**Difference**	**Error %**
Avg.	0.7157017	0.67801	0.0376917	3%(almost)

TABLE VII. CLUSTERING COEFFICIENT COMPARISON.

	Before	**After**
Clustering Coefficient Avg.	0.247242735478	0.632683188902

G. Comparison Assessment

In addition, to evaluate the results more accurately, we calculated the number of true positives (nTP) and the number of false-positive (nFP). More formally in Equation 9.



(a) Clustering Coefficient Frequency for Each Node at First Case (Before Processing).

(b) Clustering Coefficient Frequency for Each Node at Second Case (After Processing).

Fig. 5. Clustering Coefficient Frequency for Each Node.

$$\textbf{Precision value:} \frac{nTP}{(nTP + nFP)} \tag{9}$$

Thus the higher value gives better precision. Moreover, the true positive (TP) rate refers to how many real friendships have been accurately suggested by the recommended system. Similarly, the false positive (FP) rate refers to the number of actual friendships the system has not suggested. The confusion matrix is used to calculate these rates. In Table VIII shows the confusion matrix.

We have calculated multiple comparisons for our dataset. Considering the threshold value of the real dataset, the false positive rate is 0.00260085, and the true positive rate is 0.0165362. Moreover, 0.96566 is the precision value. That indicates 97% users on our dataset are recommended accurately by our proposed system.

Moreover, to compare the proposed model with other existing methods, we have re-implemented the models used in [8, 10, 11, 12, 13, 14, 16] as accordance with the description in the paper to make a fair comparison. All the models are evaluated on the same data set to ensure the fairness of the comparison. In Table IX shows a comparison of our system with some existing works that we have re-implemented on the same data set that we have used.

Graphical comparisons of proposed model with [8], [10], [11], [12], [13], [14], and [16] has shown in Fig. 6.

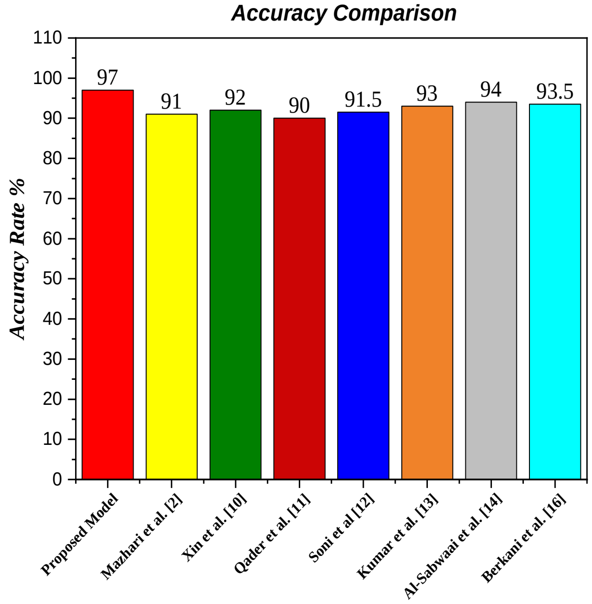


Fig. 6. Graphical Comparison of Proposed Model with [8], [10] [11], [12], [13], [14], and [16] which is Re-implemented on our Data Sets.

The ROC curve of Fig. 7 shows the true positive and the false positive rate for different threshold value.

V. Conclusion

This paper's fundamental achievement is designing and developing a user recommendation system consequent to profile attributes and network connection. This paper proposes an effective method to calculate the maximum number of similar users from a social network. Sometimes profile attributes of the user are hidden or missed, but this hidden or missed attributes data can affect profile similarity calculation. Most of the existing methods are failed to solve this problem, but our proposed approach solves it and gives better performance. Our proposed recommendation method achieved 97% accuracy and 96.566% precision which means the system properly recommended future friends for the user in the social network.

In the future, we will spread this work by combining attributes, for example, shares, comments, likes, pictures, status,

TABLE VIII. CONFUSION MATRIX

		Predicted Class		
		Positive	Negative	Total
Actual Class	Positive	True Positive (TP)	False Positive (FN)	$P = TP + FN$
	Negative	False Negative(FP)	True Negative(TN)	$N = FP + TN$
	Total	$P' = TP + FP$	$N' = FN + TN$	$S' = P + N = P' + N'$

TABLE IX. ACCURACY COMPARISON

Authors	**Year**	**Accuracy Rate %**
Mazhari et al. [8]	2015	91%
Xin et al. [10]	2020	92%
Qader et al. [11]	2020	90%
Soni et al. [12]	2020	91.5%
Kumar et al. [13]	2018	93%
Al-Sabaawi et al. [14]	2018	94%
Berkani et al. [16]	2021	93.5%
Proposed Method	-	**97%**

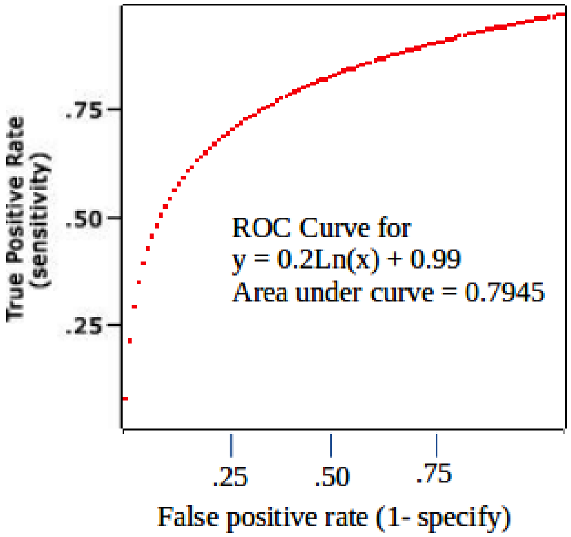


Fig. 7. ROC Curve of our Proposed FMS Algorithm.

etc. By combining those kinds of attributes with our method, we will be capable of calculating the sentiment analysis of OSN users and quickly identify and observe the illegal activities in the online social network.

REFERENCES

[1] C.G. Akcora, B. Carminati, E. Ferrari, User similarities on social networks, Social Network Analysis and Mining 3 (3) (2013) 475–495.

[2] M. McPherson, L. Smith-Lovin, J.M. Cook, Birds of a feather: Homophily in social networks, Annual review of sociology 27 (1) (2001) 415–444.

[3] T. Kaya, H. Bicen, The effects of social media on students' behaviors; facebook as a case study, Computers in Human Behavior 59 (2016) 374–379.

[4] P. Bhattacharyya, A. Garg, S.F. Wu, Analysis of user keyword similarity in online social networks, Social network analysis and mining 1 (3) (2011) 143–158.

[5] D. Yang, C. Huang, M. Wang, A social recommender system by combining social network and sentiment similarity: A case study of healthcare, Journal of Information Science (2016) 0165551516657712.

[6] R. Buettner, Predicting user behavior in electronic markets based on personality-mining in large online social networks, Electronic Markets (2016) 1–19.

[7] C. Cai, H. Xu, A topic sentiment based method for friend recommendation in online social networks via matrix factorization, Journal of Visual Communication and Image Representation 65 (2019) 102657.

[8] S. Mazhari, S.M. Fakhrahmad, H. Sadeghbeygi, A user-profile-based friendship recommendation solution in social networks, Journal of Information Science 41 (3) (2015) 284–295.

[9] Y. Yang, J. Pei, A. Al-Barakati, Measuring in-network node similarity based on neighborhoods: a unified parametric approach, Knowledge and Information Systems (2017) 1–28.

[10] M. Xin, L. Wu, Using multi-features to partition users for friends recommendation in location based social network, Information Processing & Management 57 (1) (2020) 102125.

[11] S.A. Qader, A.R. Abbas, Dual-stage social friend recommendation system based on user interests, Iraqi Journal of Science (2020) 1759–1772.

[12] M.T.S.S. Soni, P. Singhai, Friend recommendation system using machine learning method, Journal of Scientific Research & Engineering Trends, vol. 6, Issue 5, 2020.

[13] P. Kumar, G.R.M. Reddy, Friendship recommendation system using topological structure of social networks, in: Progress in Intelligent Computing Techniques: Theory, Practice, and Applications, Springer, 2018, pp. 237–246.

[14] A.M.A. Al-Sabaawi, H. Karacan, Y.E. Yenice, Exploiting implicit social relationships via dimension reduction to improve recommendation system performance, PloS one 15 (4) (2020) e0231457.

[15] M. Shabaz, U. Garg, Shabaz–urvashi link prediction (sulp): A novel approach to predict future friends in a social network, Journal of Creative Communications 16 (1) (2021) 27–44.

[16] L. Berkani, S. Belkacem, M. Ouafi, A. Guessoum, Recommendation of users in social networks: A semantic and social based classification approach, Expert Systems 38 (2) (2021) e12634.

[17] G. Razis, I. Anagnostopoulos, Discovering similar twitter accounts using semantics, Engineering Applications of Artificial Intelligence 51 (2016) 37–49.

[18] J. Lindamood, R. Heatherly, M. Kantarcioglu, B. Thuraisingham, Inferring private information using social network data, in: Proceedings of the 18th international

conference on World wide web, ACM, 2009, pp. 1145–1146.

[19] E. Brill, R.C. Moore, An improved error model for noisy channel spelling correction, in: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, 2000, pp. 286–293.

[20] B. Cao, Y. Li, J. Yin, Measuring similarity between graphs based on the levenshtein distance, Appl. Math 7 (1L) (2013) 169–175.

[21] M. Gardner, Taxicab geometry, in: The Last Recreations, Springer, 1997, pp. 159–175.

[22] M.A. Islam, L. Islam, Calculation of client similarities in large-scale on social network using recommendation framework, in: 2019 5th International Conference on Advances in Electrical Engineering (ICAEE), IEEE, 2019, pp. 679–684.

[23] J. Leskovec, A. Krevl, SNAP Datasets: Stanford large network dataset collection, `http://snap.stanford.edu/data` (Nov. 2020).