Exploring Efficiency of Data Mining Techniques for Missing Link in Online Social Network

Chainarong Sirisup
Department of Computer Science
Faculty of Science and Technology
Thammasat University
Pathumthani, Thailand
chainarong.sirisup@gmail.com

Pokpong Songmuang
Department of Computer Science
Faculty of Science and Technology
Thammasat University
Pathumthani, Thailand
pokpongs@tu.ac.th

***Abstract*—Missing link in Online Social Network (OSN) is an interesting problem for capturing missing relation and understanding user's behavior. The existing work introduced social features for training predictive models, but they used only SVM prediction technique for solving the problem. However, we suspect that other prediction techniques may give better performance. This study investigates prediction performances of SVM, k-NN, Decision Tree, Neural Networks, Naïve Bayes, Logistic Regression and Random Forest using two OSN datasets (high-density and low density). We realize that the Random Forest technique has the best performance with F1-measure score. Moreover, this technique is most robust technique for the both datasets.**

Keywords—missing link, link prediction, online social network, prediction technique

I. INTRODUCTION

Online Social Network (OSN) is an online service connecting people through the Internet. At this moment, OSN is a popular service having significant influence over people's lifestyle such as communication, public relation and advertising. Users who use OSN can communicate quickly and free of charge. The examples of popular OSN providers are Facebook, Twitter, Instagram, etc. According to a global survey of OSN users, Statista [1] found that Facebook is the most popular OSN provider. In April 2018, Facebook is the most popular OSN provider and it has approximately 2,234,000,000 users accounted for 17% of total users among the 20 OSN providers. Therefore, we select Facebook as the OSN to be used in research and development.

In general, OSN structure consists of a node representing an individual user and a link between two nodes representing the relationship between two users. For examples, a link represents *a friend relationship* in Facebook and a link represents *a follower-followee relationship* in Twitter and Instagram. At present, there are researches [2,4,5] on link relation called a link prediction, they trained their link prediction models using datasets and then they predict a link between two nodes that should have a link. Also a link that should have in OSN but it disappears, called a missing link. In reality, Facebook and Instagram use a link prediction technique to suggest friends called recommended system. Missing link analysis is applicable to a wide variety of applications. In the field of criminal investigations, missing link analysis is used to investigate more relevant suspects by considering the relationships between a criminal and any person who is probable a suspect in the criminal network. [2]

Han X. et al. [4] proposed a link prediction for new user in OSN. They used a prediction model with feature-based prediction, and all selected features influenced on the model performance. The research focused only on Support Vector Machine (SVM) technique for link prediction. Nevertheless, they lack of comparing the performances among prediction techniques and using only one OSN dataset. So, it's just only one aspect on dataset dimension.

This research aims to compare the efficiency of prediction techniques for missing link problem to reveal the best performance technique. Many prediction techniques are used to approach this problem including SVM, k-NN, Decision Tree, Neural Network, Naïve Bayes, Logistic Regression and Random Forest. We divide the experiment in to two parts as follows: 1) training prediction model for finding optimal parameter values and 2) comparison of prediction performance. Furthermore, for model tolerance testing, we use two different density's OSN datasets to compare prediction performance.

Related work is described in section II and the methods are explained in detail in section III. Section IV presents the experiments setup. The results are described in section V. Finally, section VI provides the conclusions.

II. RELATED WORK

Han X. et al. [4] proposed a friends recommended model for new users on online social network. They considered relationships between users with personal qualifications such as workplace, school, hometown etc. In order to recommend friends to new user, they computed friend probability from existing data on online social network and utilized Support Vector Machine (SVM) technique for predicting the link relations. From personal qualifications, they computed and extracted three features for training and testing i.e. Basic Feature, Derived Feature and Latent Relation Feature. Furthermore, they evaluated model with large Facebook dataset (479,000 users) and achieved the following results. First, they compared prediction performance with divided into four models as (a) Blink model - only considers basic social feature which including number of common attribute and binary similarity (b) Dlink model - only considers derived social feature (c) BDlink model – combines basic and derived social feature together and (d) BDRlink model - combines basic social feature, derived social feature and latent relation feature together. They found that BDRlink model had highest performance with AUC equaled to 0.83. In short, latent relation feature played an important role in predicting the link relationships. Second, to investigate whether the social features leveraged in BDRlink model, they used 'leave-one-feature-out' strategy (which remove one feature from all features) to train the model. They found that the performance decreases drastically if the latent

relation feature was removed. Consequently, these confirm that the latent relation feature importance for online social network prediction. And the last, to validate that BDRlink model can predict link relations more accurately if new users provide more attributes. Found that, the prediction accuracy would increase if new users provide more attributes.

Moreover, several publications [6,7] introduced the prediction techniques for approach prediction problem such as SVM, k-Nearest Neighbor (k-NN), Decision Tree, Neural Networks, Naïve Bayes, Logistic Regression and Random Forest. Moreover, the use of different prediction may lead to different performances. For instance, Hasan M. et al. [10] studied topic of link prediction by using Supervised Learning. They extracted and identified features for supervised learning. They considered various prediction techniques such as Decision Tree, k-NN, Multilayer perceptron, SVM and Neural Network to set experimental. Then, they compared all of prediction techniques performance by using various metrics such as accuracy, precision-recall, F-values, squared error etc. with 5-folds cross validation. Finally, they found that most of prediction techniques were good performance, but SVM with narrow margin defeated all of them. Besides they ranked features and found a small subset of features that play significant role in link prediction problem.

Therefore, SVM technique may not be the best technique for prediction in missing link problem, so, we propose the experiments for comparison prediction techniques (including SVM, k-NN, Decision Tree, Neural Network, Naïve Bayes, Logistic Regression and Random Forest) and finding the best technique. Moreover, we use two dataset with different density value to compare performance between prediction techniques and density value. Besides, the experiments are described in the next section.

III. METHODS

To carry out the research goal, we propose dataset acquisition, feature creation and prediction technique for approach the missing link problem in this section.

A. Data description

To gather the experimental dataset, we use Facebook API for collect a user's public information on Facebook. For all public user data collection, we collect user's information such as friendship, demographic and interests for social attributes analysis. Finally, we consider ten different social attributes including age, gender, school, university, hometown, current city, works and user's favorites (music, movie and TV show).

We collect two datasets on Facebook with different density and compute density value as shown in TABLE. I. In addition, Hoppe B. et.al. [15] proposed density computation method for social network analysis as shown in equation 1. Density value is a proportion of number of existing links in the network and maximum possible links which could exist in the network. They used number of nodes and links for computing density value where M refers to number of links and N refer to number of nodes. In case of OSN is a complete graph, a density value is the maximum value as 1.

Moreover, to define missing links in experimental, we random remove 10% of the links from the datasets as a missing links.

$$Density = \frac{2M}{N(N-1)} \quad (1)$$

TABLE I. DATASET'S DENSITY VALUE

Dataset	No. Nodes	No. Links	Density Value
Dataset 1	200	926	0.05
Dataset 2	180	558	0.03

B. Features

In Han's work [4], they described social attributes for formulating the prediction model with high performance. They established features with the social attributes and conducted the study based on real social dataset. Hence, we applied their feature creation technique to create our feature sets with the following:

1) Basic features: The basic features are consistent with some of user's common social attributes. There are two types of basic features:

- *Binary similarity* calculated from comparison between two users on each attributes. For each attribute, its value equals 1 if the attribute have the same value; otherwise, it is 0.
- *Number of common attributes (NCA)* calculated from summation of all binary similarity attributes value.

2) Derived features : The derived features are the feature that considered two user's social attributes such as following:

- *Attribute distance* is the indicated familiarity between two considering users. It can be calculated from a distance between two user's age and users' geometric distributes (current city and hometown).
- *Attribute correlation* represented relation probability between users. For instance, users who have same attribute value such as work, current city and hometown will have a higher probability than those who don't.
- *Interest similarity* represented closeness between two users. It can be calculated from the similarity between two users' favorites (music, movie and TV show). The previous work [3,4] used cosine similarity to approach this feature.

3) Latent relation score feature: The latent relation score feature representes friend probability between user "u" and friends of user "v". As shown in Fig 1, user "u" and "v" will be a friend or linked together, if "u" and friend of "v" shared similar atrributes. For instance, there are four shared attributes and 4 unshared attributes between user "u", user "w_1" and user "w_2". In addition, the previous study [4] introduced equation for computing the latent relation score as following equation 2.

$$Latent\ relation\ score = \frac{1}{1 + e^{-\beta(r-\alpha q)}} \quad (2)$$

Where, r is a number of shared attributes called latent link, q is a number of unshared attributes called disconnection, β is an exponential regulator, and α is regulator for punish value.

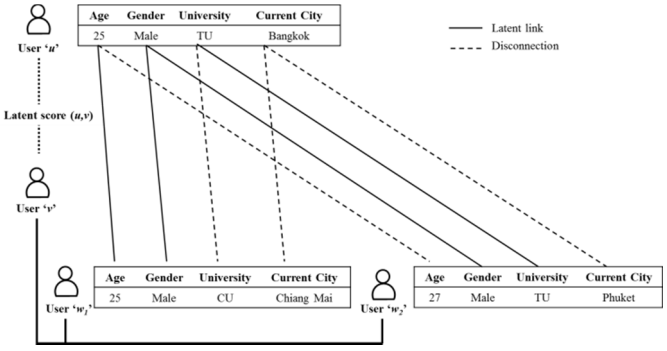


Fig. 1. Latent relations between user "u" and user "v"

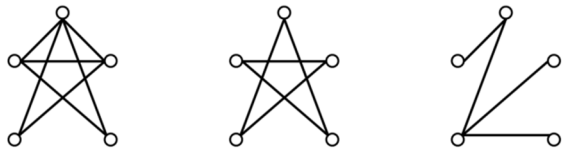
Finally, total features are twenty-one features including: binary similarity features (age, gender, school, university, hometown, current city, work, user's music favorite, user's movie favorite and user's TV show favorite), NCA feature, age distance feature, hometown distance feature, current city distance feature, hometown correlation feature, current city correlation feature, work correlation feature, music's interest similarity, movie's interest similarity, TV show's interest similarity and latent relation score feature.

We consider all relationship (e.g. friendship or non-friendship) between all users in social network dataset and combine them with dataset features. So, there is one more feature to consider relationship that is label feature (1 for friendship and 0 for non-friendship). Furthermore, data normalization is a necessary process to model formulation due to a variety of scale values in dataset. Thus, we normalize all values to 0-1 range.

C. Performance evaluation metrics

We compare performance of the aforementioned techniques using different performance metrics as the following:

Where, TP stands for true positives (prediction result has link and actual has link too), FP stands for false positives (prediction result has link but actual doesn't has link), TN stands for true negatives (prediction result doesn't has link and actual doesn't has link too) and FN stands for false negatives (prediction result doesn't has link but actual has link). For example as show in Fig. 2, there are three relation networks that are illustrated. Fig. 2(a) is a complete network that represent to original dataset. Then, Fig. 2(b) is the link removing network that it's received from random remove links in the complete network. Finally, in Fig. 2(c) represent the prediction result. As above-mentioned, we can compute TP, FP, TN and FN values as following: TP is equal 3, FP is equal 1, TN is equal 2 and FN is equal 3. Moreover, we use these values to calculate other performance metrics such as accuracy, precision, recall and F1-measure. So, the other performance metrics can be described as below.



(a) Complete network (b) Link removing network (c) Prediction result network

Fig.2. Illustration of network relation : (a) Complete network. (b) Link removing network. (c) Prediction result network

- *Percentage of link recovers:* In our experiment, we removed 10% of link relations (friends) between users and blend them into the dataset. After, we trained and evaluated the models, percentage of link recovers can compute by equation 3.

$$\%\ Link\ recovers = \frac{No.\ Link\ recovers}{No.\ Link\ removes} x\ 100 \quad (3)$$

- *Accuracy:*

$$Accuracy = \frac{|TP| + |TN|}{|TP| + |FP| + |TN| + |FN|} \quad (4)$$

- *Recall:*

$$Recall = \frac{|TP|}{|TP| + |FN|} \quad (5)$$

- *Precision:*

$$Precision = \frac{|TP|}{|TP| + |FP|} \quad (6)$$

- *F1-measure:*

$$F1 - measure = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (7)$$

- *AUC(Area under the ROC Curve):*

  In addition, the Receiver Operating Characteristic or ROC Curve is a metric which represents performance tradeoff between true positives and false positives at different thresholds.[8] In order to compare the prediction techniques, we have to reduce the ROC Curve performance representing from two-dimension to one-dimension. A common method is to calculate the area under the ROC Curve namely AUC. [9]

IV. Experiments Setup

The previous studies in term of prediction techniques were used to consider for missing link problem. We use seven prediction techniques consist of SVM, k-NN, Decision Tree, Neural Networks, Naïve Bayes, Logistic Regression and Random Forest. We divide the experiment in to two parts as follows.

A. Training prediction model for finding optimal parameter values

Due to k-NN, Neural Networks and Random Forest techniques are required parameters adjustment for model training, so, we set up experiments in order to determine the optimal parameter values as.

1) Optimal k value in k-NN technique: Many previous studies typically arbitrary chose k values. For instance, Ma C. et al. [11] selected number of k as odd number and less than 65 for demonstration. Wang J. et al. [12] pre-selected number of k from 1 to 50. Since there are no specific rule to select number of k value, we would pre-select number of k value as odd number and less than 50 for the experiment. We use 10-folds cross validation and accuracy prediction to compute prediction performance for each k value.

2) Optimal number of hidden nodes in Neural Network technique: Determining optimal number of hidden nodes is the most challenging aspect of Neural Network design. There're several methods for approaching this problem, e.g., exhaustive finding by setting experiment. [13] Moreover, Panchal F.S. and Panchal M. introduced method by varying number of hidden layers and number of hidden nodes.[14] We set up experiment by vary number of hidden layers and number of hidden nodes. For hidden layer (HL), we vary number of HL as 1 and 2. So, HL1 is a first hidden layer and HL2 is a second hidden layer. For hidden node (HN), we vary number of HN as 0-10. For instance, HL1-1&HL2-0 means the first hidden layer contains one hidden node and the second hidden layer not contains any hidden node. Then, we use 10-folds cross validation and accuracy prediction to compare performances.

3) Optimal number of tree in Random Forest technique: The number of tree in Random Forest is a required parameter setting, so we have setup experiment to finding the optimal value. Previously, Norouzi M. et.al. [16] and Cuzzocrea A. et.al.[17] used number of tree variation in order to find the best accuracy in their experiments. Since there are no specific rule to select number of tree value, then we pre-select number of tree values as 10, 50, 100, 500 and 1,000. Then, we use 10-folds cross validation and accuracy prediction to compare performances.

B. *Comparison of prediction technique performance with diferent OSN density*

After we get optimal parameter values for the prediction techniques, we compare the performance of seven prediction techniques including SVM (construct with linear separability solver), k-NN (determine k parameter as optimal value from experiment), Decision Tree (construct with Classification and Regression Trees algorithm :CART), Neural Networks (determine hidden layer as optimal value from experiment), Naïve Bayes, Logistic Regression (determine penalty as L2) and Random Forest (determine number of tree as 500). All prediction techniques are trained with OSN dataset containing twenty-one features and one label feature from section III-B. We apply all prediction with Sklearn library in python programming language and evaluate the performance with 10-fold cross validation and some metrics from section III-C. Furthermore, we set up an experiment to evaluate the prediction performance with different OSN density (high-density and low density).

V. RESULTS

The experiments are performed on OSN dataset in which twenty-one features are taken. Two dataset with different density were used for comparison. The features in OSN dataset consisted of binary similarity features (age, gender, school, university, hometown, current city, work, user's music favorite, user's movie favorite and user's TV show favorite), NCA feature, age distance feature, hometown distance feature, current city distance feature, hometown correlation feature, current city correlation feature, work correlation feature, music's interest similarity, movie's interest similarity, TV show's interest similarity and latent relation score feature. Several prediction techniques are used to compare the performance in this experiment. In terms of optimal parameters for some techniques including k-NN, Neural Network and Random Forest, we set up experiments for approach them and results show below.

A. *Training prediction model for finding optimal parameter values*

*1) Optimal k value in k-NN technique:*For the optimal k value in k-NN technique, the result represent k equal 1 effect to the best F1-measure which can be seen in Fig. 3. Thus, we use this optimal k value for train k-NN technique in next experiment.

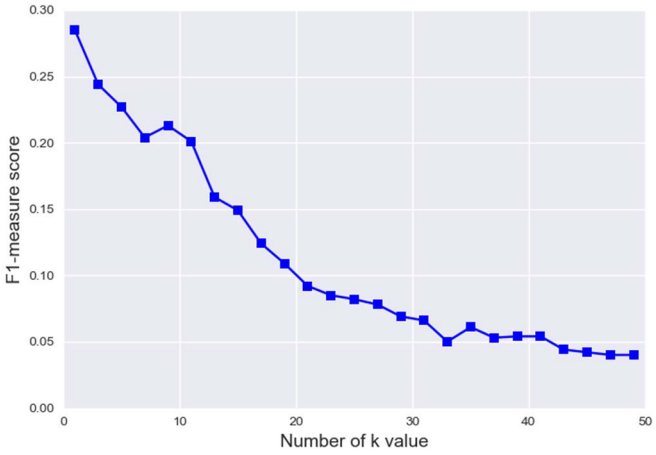


Fig. 3. k-NN F1-measure score for different k value base on, 10-folds cross validation, k is odd and less than 50.

*2) Optimal number of hidden nodes in Neural Network technique:*For the optimal number of hidden layer and hidden nodes in Neural Network technique, the result represente the optimal values are 1 and 10 for the number of hidden layer and the number of hidden nodes, respectively. So, we will use this value for training Neural Network in next experiment. As depicted in Fig. 4.

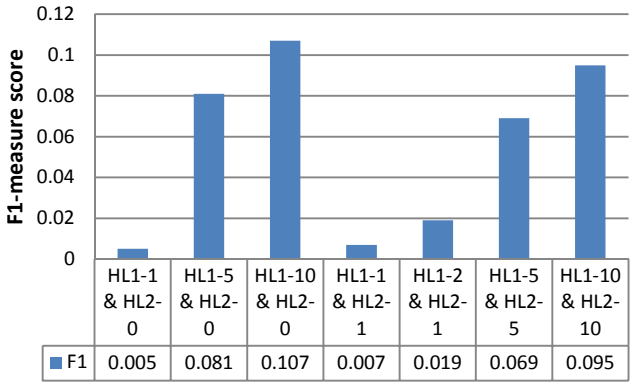


Fig. 4. Neural Network F1-measure score for different number of hidden layer and number of hidden node base on, 10-folds cross validation.

3) Optimal number of tree in Random Forest technique: For the optimal number of tree in Random Forest technique, the result represente the number of tree as 500 is the best performance as show in TABLE II. So, we will use this value for training Random Forest in next experiment.

TABLE II. RANDOM FOREST PERFORMANCE FOR DIFFERNCE NUMBER OF TREE PARAMETER

Evaluation Metric	Number of tree				
	10	**50**	**100**	**500**	**1000**
F1-Measure	0.374	0.386	0.397	**0.402**	0.399

B. *Comparison of prediction technique performance with diferent OSN density*

In addition, we get the optimal parameter values for k-NN, Neural Network and Random Forest technique. Then, seven prediction techniques containing SVM, k-NN (k=1), Decision Tree, Neural Networks (HL1-10 and HL2-0), Naïve Bayes, Logistic Regression and Random Forest (number of tree as 500) are compared performance with different OSN datasets including dataset 1 (density value as 0.05) and dataset 2 (density value as 0.03). The evaluation metrics (percentage of number link recovers, accuracy, precision, recall, F1-measure and AUC) are used to comparison.

For dataset 1 (a high-density OSN dataset), the result demonstrates that Random Forest technique is the best performance on F1-measure equal to 0.40 as shown in TABLE III. Another part for dataset 2 (a low-density OSN dataset), the result demonstrates that Random Forest and Decision Tree technique are the best performance on F1-measure equal to 0.34 as shown in TABLE IV.

Due to the dataset 1 has a high-density OSN dataset, so, the feature distribution is weak and sparky features. Lastly, the complexity prediction technique like Random Forest gets the best performance. Nevertheless, the dataset 2 has a low-density OSN dataset, then the feature distribution is tight and strong features. So, Decision Tree technique can perform efficiency like Random Forest technique. Finally, the robust technique for missing link prediction is also the Random Forest technique due to it has a best performance in both of two dataset.

TABLE III. PERFORMANCE COMPARISON OF PREDICTION TECHNIQUES ON DATASET 1

Prediction techniques	**10-folds cross validation**					
	% Link Recovers	***Accuracy***	***Precision***	***Recall***	***F1-measure***	***AUC***
	%	***Avg. (SD.)***	***Avg. (SD.)***	***Avg. (SD.)***	***Avg. (SD.)***	***Avg. (SD.)***
SVM	0%	0.96 (±0.03)	0.00 (±0.00)	0.00 (±0.00)	0.00 (±0.00)	0.52 (±0.06)
k-NN (k=1)	51.08%	0.92 (±0.01)	0.24 (±0.15)	0.44 (±0.09)	0.28 (±0.15)	0.65 (±0.03)
Decision Tree	44.62%	0.95 (±0.01)	0.39 (±0.20)	0.43 (±0.09)	0.38 (±0.140	0.68 (±0.04)
Neural Networks (HL1-10 & HL2-0)	5.38%	0.96 (±0.03)	0.40 (±0.19)	0.07 (±0.03)	0.11 (±0.05)	0.78 (±0.03)
Naïve Bayes	32.26%	0.86 (±0.04)	0.12 (±0.08)	0.37 (±0.08)	0.17 (±0.10)	0.67 (±0.05)
Logistic Regression	2.15%	0.96 (±0.03)	0.37 (±0.32)	0.01 (±0.01)	0.03 (±0.03)	0.67 (±0.06)
Random forest (no.tree=500)	35.48%	0.97 (±0.02)	0.58 (±0.24)	0.33 (±0.08)	**0.40 (±0.12)**	**0.89 (±0.04)**

TABLE IV. PERFORMANCE COMPARISON OF PREDICTION TECHNIQUES ON DATASET 2

Prediction techniques	**10-folds cross validation**					
	% Link Recovers	***Accuracy***	***Precision***	***Recall***	***F1-measure***	***AUC***
	%	***Avg. (SD.)***	***Avg. (SD.)***	***Avg. (SD.)***	***Avg. (SD.)***	***Avg. (SD.)***
SVM	0%	0.97 (±0.02)	0.00 (±0.00)	0.00 (±0.00)	0.00 (±0.00)	0.50 (±0.08)
k-NN (k=1)	47.32%	0.91 (±0.01)	0.16 (±0.11)	0.38 (±0.08)	0.20 (±0.12)	0.66 (±0.03)
Decision Tree	41.96%	0.96 (±0.01)	0.37 (±0.23)	0.37 (±0.10)	**0.34 (±0.16)**	0.70 (±0.06)
Neural Networks (HL1-10 & HL2-0)	6.25%	0.97 (±0.02)	0.44 (±0.20)	0.11 (±0.05)	0.16 (±0.07)	0.78 (±0.03)
Naïve Bayes	49.11%	0.87 (±0.03)	0.10 (±0.07)	0.38 (±0.08)	0.14 (±0.10)	0.69 (±0.05)
Logistic Regression	2.68%	0.97 (±0.02)	0.57 (±0.35)	0.04 (±0.03)	0.07 (±0.05)	0.70 (±0.07)
Random forest (no.tree=500)	31.25%	0.97 (±0.02)	0.56 (±0.26)	0.26 (±0.09)	**0.34 (±0.12)**	**0.90 (±0.04)**

VI. Conclusions

In this research, different prediction techniques such as SVM, k-NN, Decision Tree, Neural Networks, Naïve Bayes, Logistic Regression and Random forest are used to compare the performance for missing link problem in OSN. The Random Forest technique has superior perform over other prediction techniques. Furthermore, the Random Forest technique is the robust technique for missing link prediction in the both of two OSN dataset (high-density and low-density). In the future, this work can be extended to improve the prediction performance of Random Forest for the missing link problem.

References

[1] Statista. (2018, April). Most famous social network sites worldwide as of April 2018 ranked by number of active users. . Retrieved from https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/

[2] Berlusconi, G., Calderoni, F., Parolini, N., Verani, M., & Piccardi, C. (2016). Link Prediction in Criminal Networks: A Tool for Criminal Intelligence Analysis. PLoS ONE 11(4).

[3] X. Han, L. Wang, S. Park, A. Cuevas, and N. Crespi, (2014). Alike People, Alike Interests? A Large-scale Study on Interest Similarity in Social Networks. International Conference on Advances in Social Networks Analysis and Mining: 491-496.

[4] Han, X., Wang, L., Han, S. N., Chen, C., Crespi, N., & Farahbakhsh, R. (2015). Link Prediction for New Users in Social Networks. IEEE International Conference on Communications (ICC), 1250-1255.

[5] Peng, W., BaoWen, X., YuRong, W., & XiaoYu, Z. (2015). Link Prediction in Social Networks: the State-of-the-Art. Sci China Inf Sci. 2015; (58).

[6] Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. Informatica 31:249–268.

[7] Kumari, R., & Srivastava, S. (2017). Machine Learning: A Review on Binary Classification. International Journal of Computer Applications 160(7): 11-15.

[8] Yang, Y., Lichtenwalter, R. N., & Chawla, N. V. (2015). Evaluating link prediction methods. Knowl Inf Syst 45: 751-782.

[9] Fawcett, T. (2006) An introduction to ROC analysis. Pattern Recognition Letters 27: 861-874.

[10] Hasan, M. A., Chaoji, V., Salem, S., & Zaki, M. (2006). Link Prediction using Supervised Learning. Proceedings of SDM 06 Workshop on Link Analysis, Counterterrorism and Security.

[11] Ma, C., Yang, W., & Cheng, B. (2014). How the Parameters of K-nearest Neighbor Algorithm Impact on the Best Classification Accuracy: In Case of Parkinson Dataset. Journal of Applied Sciences 14(2): 171-176.

[12] Wang, J., Neskovic, P., & Cooper, L. N. (2007). Improving nearest neighbor rule with a simple adaptive distance measure. Pattern Recognition Letters 28: 207-213.

[13] Thomas, A. J., Petridis, M., Walters, S. D., Gheytassi, S. M., & Morgan, R. E. (2015). On Predicting the Optimal Number of Hidden Nodes. International Conference on Computational Science and Computational Intelligence: 565-570.

[14] Panchal, F. S., & Panchal, M. (2014). Review on Methods of Selecting Number of Hidden Nodes in Artificial Neural Network. International Journal of Computer Science and Mobile Computing 3(11): 455-464.

[15] Hoppe, B., & Reinelt, C. (2010). Social network analysis and the evaluation of leadership networks. The Leadership Quarterly 21: 600-619.

[16] Norouzi, M., Collins, M. D., Fleet, D. J., & Kohli, P. (2015). CO2 Forest: Improved Random Forest by Continuous Optimization of Oblique Splits. 1506.06155.

[17] Cuzzocrea, A., Francis, S., & Gaber, M. (2013). An Information-Theoretic Approach for Setting the Optimal Number of Decision Trees in Random Forests. International Conference on Systems, Man, and Cybernetics:1013-1019.