International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering
Vol. 5, No. 1, 2018, pp. 222-240.
ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

Improving Link Prediction in Dynamic Co-authorship Social Networks

Atefeh Latif [a], Alireza Hedayati [b], Vahe Aghazarian [c]

[a] *Computer engineering graduate student, Islamic Azad University, Central Tehran Branch, Tehran, Iran.*

[b] *Assistant Professor and Faculty Member Computer Group, Islamic Azad University, Central Tehran Branch, Tehran, Iran.*

[c] *Assistant Professor and Faculty Member Computer Group, Islamic Azad University, Central Tehran Branch, Tehran,Iran.*

Abstract

The current study is based on a type of social cooperation networks called co-authorship network which its related activities revolve around authors' citations and their mutual cooperation in researches. Cooperation networks are established to better achieve consistent goals or common goals, and interactions are supported by computers in these networks. Cooperation social networks' order and system focus on their structure, behavior and dynamicity. The current study aims at improving co-authorship social network links prediction by the use of RapidMiner data mining software. To do this, supervised learning algorithms were used. In comparison to the similar cases, it was concluded that link prediction accuracy is 98.63% and the error rate of the method has decreased in comparition with all other methods in this search. The average of improvement in error rate is 47%.

Keywords: Social Networks, Co-authorship, Supervised Learning, Link Prediction

Introduction:

Cooperation is one way to improve the quality of science. Co-authoring is a clear form of cooperation and for this reason the aim of studying science and bibliometric is cooperation. Cooperation between scientists in recent decades is increasing and the widespread availability of information technology, network information, national facilities and international academic cooperation has improved. Cooperation increases scientific and research potential of the country, although cooperation is not considered as a quality indicator, but is a means to improve the scientific literature quality [1]. As we know, a network of scientists where the relationship between the two scientists is established by their joint written in one or several scientific papers is called "dependence network", where activists are linked to a group of authors of an article through their common membership. They are more social than dependence networks. Many people who have written an article together, in fact, are familiar with each

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

other. Of course, there are exceptions. However, this network represents a real professional interaction between scientists and may be the largest social network that has ever been studied.

Much of routes between other authors and scientists are shown in the Network through just one or two partners. Rapprochement Criterion for joint relations includes the number of articles written by a pair of scientists. The criterion considers the number of other authors who have written an article with them. Using this criterion, researchers have added weight to cooperation networks and use the networks to find those scientists who have the shortest average distance with others [2]. Social networks can be used to model the interactions between individuals in a group or community. The networks can be displayed as graphs so that a vertex is considered as a person in the group and an edge indicates a relationship between corresponding people. A relationship usually is formed with mutual interests. Social networks are very dynamic, because over time new edges and nodes are added to the graph. Understanding dynamisms that are arising from the growth of social networking is difficult because of the large number of variable parameters. But, one easier thing is to understand the connection between certain nodes [3].

Related Works:

Link prediction on social networks in various fields such as sociology, anthropology, information science and computer science have attracted much attention. Most existing methods have adopted a display of the static graph for predicting new links. However, this method loses some dynamic network topological important information. The researchers provided a method to predict the link in dynamic networks with the integration of Time information, the structure of society and the centrality of node in the network.
Information about all aspects is very useful to predict the potential links on social networks. Time information provides the behavior of a link in the dynamic network, while, clustering community shows how the link between two nodes is strong alone, based on whether they share the same community or not. The centrality of node that measures its relative importance in a network has a strong relationship with the future links in social networks. Researchers predicted the future importance of a node by the centrality of particular vector and used it to predict. Typological integration of information, including social structures and centralization, with Time information provide a more real model to predict link in the dynamic networks. In the future topology prediction of a dynamic social network, the fundamental problem is to identify nodes that will be placed in more important posts in the future. In various applications, single-target prediction functions are used to assess the significance of node. Because it's more likely that the nodes related to each other, the researchers provided an approach to potential predict links using the node future significance under the central matrix. Central matrix aggregates topological information throughout history.
As a result, it offers a combination of the relative importance of node over time. Because the newer important nodes will be present in the future in the network more likely than older nodes, the proposed combination assigns more weight to the rates of their centrality. In this model, eigenvector centrality was used to measure the importance of nodes in the network. Many researchers presented the link prediction as a way to identify critical nodes or edges which appear in the future. In this study, the researchers used the importance of nodes and edges to predict future in dynamic social networks. In addition, the time factor was considered.
In the proposed algorithm, the integrated time series model (ITM) and analysis of social networks provide more real information from different aspects and eliminates the influence of one-sided data. The algorithm ITM combines three types of information, the society information, node centrality information and time series data. Experimental results show that in the real data set, the proposed method based on the integrated model can effectively predict the future links on time social networks, and achieve higher quality results than traditional methods [14].

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

Link Prediction:

Link Prediction is an important measure for social network analysis, as well as has other applications in domains like data recovery, bioinformatics and e-commerce. There are different techniques for predicting link, ranging from classification based on the feature to the core-based method and matrix factorization and probabilistic graphical models. The models are different in terms of complexity of the model, prediction performance, scalability and generalization ability. Then, the link prediction has been proposed in the area of mathematics [3].

In a social network, G = <V, E> each edge e = <u, v> ∈ E is an interaction between u and v where the time t (e) shows the interaction. Multiple interactions between u and v are shown by the edges parallel with different time stamps. For two times t <t ', G [t, t'] below graph of G shows that contains all edges between t and t'. Four times t0 <t0 ^ '<t1 <t'1 and a network access algorithm G [t0, t0 '], the output should be a list of edges which are not in G [t0, t0 '] and it is predicted that they are available in the network G [t1, t1 ']. Time interval [t0, t0'] is training distance and [t1, t1 '] is test distance. Given that the social networks are grown with increasing nodes and edges, there is no prediction for edges that their two heads are not placed in the training distance. k Training and k Test parameters are used for evaluating link prediction methods. The core collection includes all nodes which have the least k Training edges in G [t0, t0 '] and at least k Test edges in G [t1, t1']. Evaluation is done to determine the accuracy of prediction of edges between the core elements [4].

In order to increase forecast accuracy, as well as the appropriate feature selection by particle swarm optimization (PSO) algorithm, t is necessary that normalization is performed on the data. For this purpose, the normalization method is used in RapidMiner tool.

The concept of normalization has different meanings in statistics. Normalization of the data or variables is considered as the simplest normalization method and is defined as a method where the data with different domain will be in the same range. In other words, data miner may be faced with situations where data features include the values which are in different domain or range. Features that have allocated large values to themselves may have much greater effect on the cost function than the features that have little value. The problem to be resolved with the features normalized so that their values are in similar domains. It is therefore necessary that numbers to be normalized in the range of zero and 1 in order to increase the accuracy of the algorithm applied on it. In this section, the PSO algorithm is described in order to select the appropriate features of the graph. It should be noted that, according to the Rapidminer data mining software special capability in the field of feature selection algorithms, the implementation of the proposed method is done using Rapidminer software tools.

First, Kennedy and Eberhart presented particle swarm optimization method. Components of group follow a simple behavior in such a way that each member of group imitates the success of its neighbors. In this algorithm, the group members move in the search space and will be gathered at the optimal point. In PSO, it is assumed that each particle is able to maintain the position discovered from the beginning until now. For example, the best position ever found by the particle swarm is called the best global position and the best position found ever by any of the particles is known as the best personal position. To find the optimal global position in the optimization problem, the particles learn of the best personal and global position. In particular, the learning mechanism in the standard PSO is as follows [5].

$$V_i(t+1) = \omega V_i(t) + C_1 R_1(t)\,(p^{best_i}(t) - X_i(t)) + C_2 R_2(t)\,(gbest(t) - X_i(t)) \qquad (1)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \qquad (2)$$

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

Each particle has a position that defines the coordinates of the particle in the search space. The Position of the particle changes with moving the particle over time,. Xi (t) determines the position of the particle i at time t. Every particle has a certain velocity to move in the space. Vi (t) indicates the velocity of the particle i at time t. ω is an inertia weight which considers a part of the particle velocity for calculating in the previous velocity of the particle. C1 and C2 are called constant which determine that impact of the best position of each particle and the best global position. R1 (t) and R2 (t) are random variables between zero and one. PSO algorithm pseudo-code is provided below.

```
Initialize All Particles Position and Speed Randomly
 and
velocities in the search space
While(termination conditions are not met)
{
  For each particle i do
Update the velocity of particle i according to equation (1).
Update the position of particle i according to equation (2).
Map the position of particle i in the solution space
and
evaluate its fitness value according to the fitness function.
      Update pbesti(t) and gbest ( t) if necessary.
  End for
}
```


Received:12 February 2018/Accepted:16 May/Published:1 June 2018

International Academic Journal of Science and Engineering,
Vol. 5, No. 1, pp. 222-240.

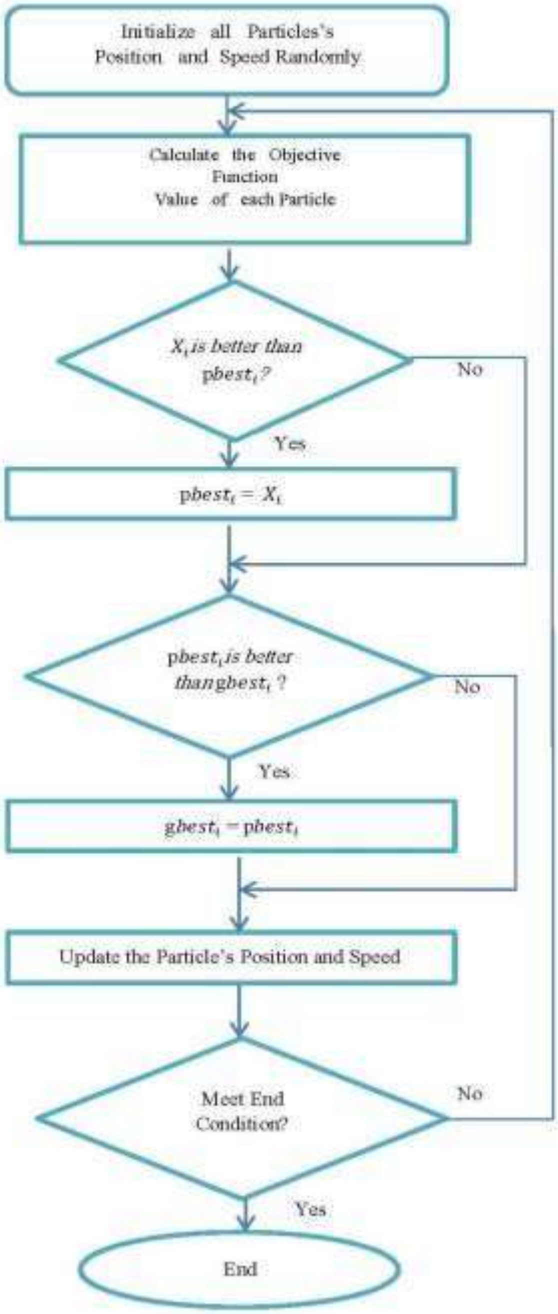

Figure 1: PSO Algorithm[8]

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

Datasets:

In this section, the dataset used in this study is studied. DataSet used is called DBLP. This dataset has two main file under the following titles:

all edge.xlsx

all node.xlsx

The data have been extracted from the database DBLP, and after reviewing more than 170 research papers from 2011 to September 2015 were extracted in two excel files. The resulting graph contains 621 nodes and 4318 edges and is directed and weighted. With increase in the edges weight, their thickness also increases. In this study, only two files with names listed above are used. The resulting graph is a heterogeneous and multi-link graph. It is heterogeneous, because there are two nodes (authors and journals) in the graph and multi-link because the two links (one link between the writers of origin and destination and the link between the writers of origin and journals in which their article has been published). Weighting is done based on the number of citations of an author in his article to another writer. In the table on the next page 50 first records for the nodes are provided as sample.

Proposed Method:

Many methods have been proposed for link prediction in co-authorship networks, some of which are called similarity-based algorithms, in fact, they are used to obtain structural similarity or distance between two nodes such as score (x, y). This function represents the amount of similarity between a pair of nodes in the social network G. One of the most important and significant point is that after calculating the neighbors of a node (using the algorithm K nearest neighbor), similarities between the current node with other nodes must be calculated and a weight or point must be considered for each of them. Then, scores are arranged in descending order, and nodes that have a high point are to be considered as related neighbors. Thus, nodes with more points can much more likely communicate with the current node at the near future. The steps of the proposed method are presented in the following as step-by-step.

First of all, the relevant data is extracted from the DBLP database, and after converting into a format acceptable, they are entered the software. Data used for implementation and predicting future link in the dynamic authorship social networks is in relation to relation between the numbers of writers and their desired publications. In order to prepare and final pre-processing on the data, that's enough that unused date and fields are removed from the dataset to make predictions with more precision and speed. At the end of this section, the data are converted into acceptable formats for data mining tools like Rapidminer.

In this section, using the appropriate clustering algorithms such as k-means, all data is clustered including nodes and edges and presented in categories. It should be noted that due to the limited number of features in the target dataset, feature selection is done according to their importance.

At this stage, k-means clustering algorithm splits nodes which communicate with each other or nodes that are lacking communication and hence a detailed analysis can be done on this section. Therefore, clustering algorithm presented classifies edges into various groups using measurement criteria. Any cluster shows potential dependence between the nodes.

In this section, the model and pseudo-code of the proposed method is shown to predict links in the future in dynamic authorship social media.

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

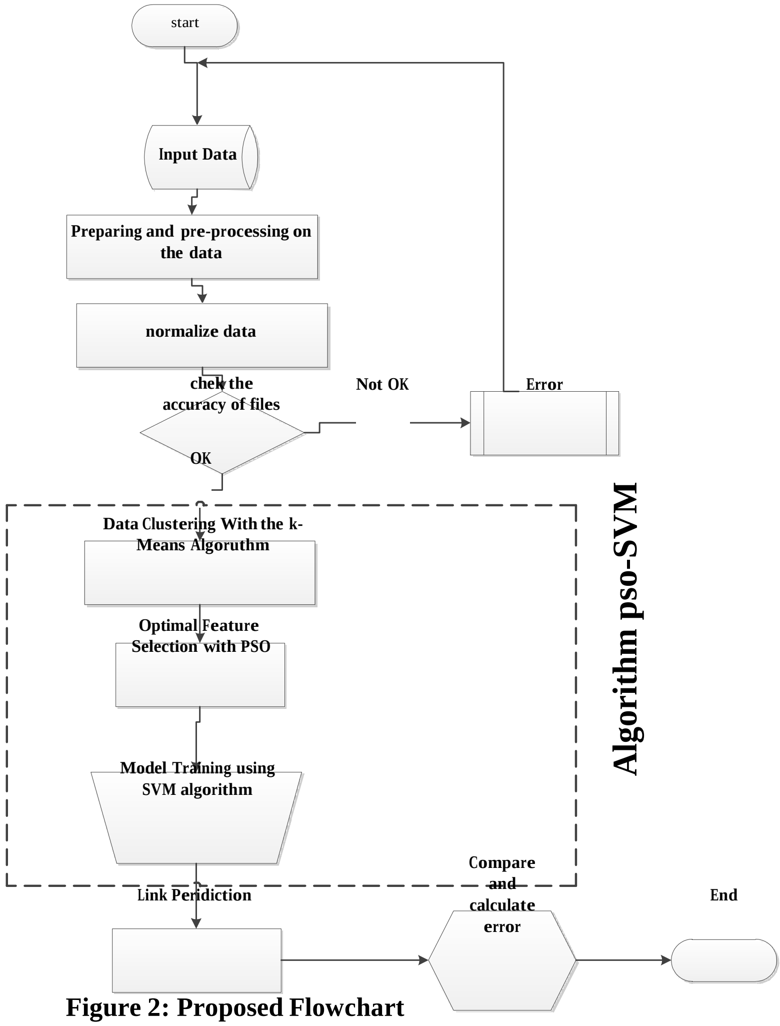

Figure 2: Proposed Flowchart

Given that the nodes are not classified and it is necessary that the best features of nodes are selected by PSO algorithm, feature extraction from DataSet is done using the PSO. 70% of the PSO algorithm output is applied as SVM algorithm input. Future links prediction is done in the future. The accuracy and error of the proposed method will be evaluated using 30% of the data.

The algorithm of the proposed method is exactly in accordance with the procedures in the flowchart. The algorithm of the proposed method is as follows.

```
Function My_Aproach(G,GTrain,GNew)
{
        Initialization:
```


Received:12 February 2018/Accepted:16 May/Published:1 June 2018

```
        Data=input Dataset;
        DataPr=PreprationData(Data);
        DataPP=Preprocessing(Data);
        Main:
        if(DataPP Is Okey)
        { Data_N=Normalize(Data_PP);
        GT=Generate_GTrain(DataPP_N);
        GN=Generate_GNew(DataPP_N);
        k-means(k,GT);
        // Tranining
        TrainingData(GT);
                NN=PSO-SVM(GT);
        NL=Calculte_Similarities(NN);
        Sort_Desc(NL);
        //Testing
        Accuracy=Evaluation(GN);
        Error=(100-Accuracy)*100;

}
else    Show Related Message;

Result:
}
```

Implementation:

The method proposed in this study is implemented using Rapidminer software. Also, the following table shows details about the system in which implementation of the proposed method and the evaluation of results is done.

Table 1: Systems Characteristics For Implementation and Evaluation of Results

Characteristics	Hardware/software
Windows7	OS
OS 64 bit	Type Of OS
5.06 Giga Byte usable- 6 Giga Byte	RAM Memory
Intell processor(– Core™)i5 CPU –)Q 720 @ 1.60GHz 1.60 GHz	Processor

Thus, according to a system with above features, the implementation is carried out and the results are evaluated. After preprocessing data in accordance with the procedures described in the beginning of the chapter, being ready to enter Rapidminer data mining software, the proposed method should be applied on it. Before applying the PSO and SVM algorithms, categories must be provided to each sample. For this purpose, K-Means clustering technique is used for clustering and determining the categories for each sample. The results of this phase are applied to the next phase which is the implementation of the PSO algorithm. After data preparation, and separation of training and testing samples in this study with the help of feature selection techniques previously described, the most prominent features that have the greatest impact on the determination of output were elected.

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

As will be seen, data is entered the model provided using Read CSV dat to select the optimal features. Then Outliers or data with no amount will be deleted. Then, the data is normalized and entered then Optimize Selection Control. After performing feature selection algorithm, it is observed that, the error is very small and feature selection is performed with acceptable accuracy. In the following figure, evaluation of PSO methods is shown in order to select the features.

Figure 3: Evaluation the Performance of the PSO Feature Selecting Algorithm

In the figure above, the root mean square error (RMSE) is calculated that is equal to 0.717%.
The square error (SE) is equal to 0.514%. So, so far the optimal properties were extracted from the data. After that, the features have been extracted, it is necessary that the hybrid method which is a combination of PSO and SVM methods to be performed on a dataset, and the necessary prediction must be implemented. In order to implement SVM model on data extracted from the previous stage (implementation of algorithm PSO), it is sufficient to provide the relevant model in the Rapidminer data mining software, and the proposed method must be applied on it. In the figure below, the model is shown.

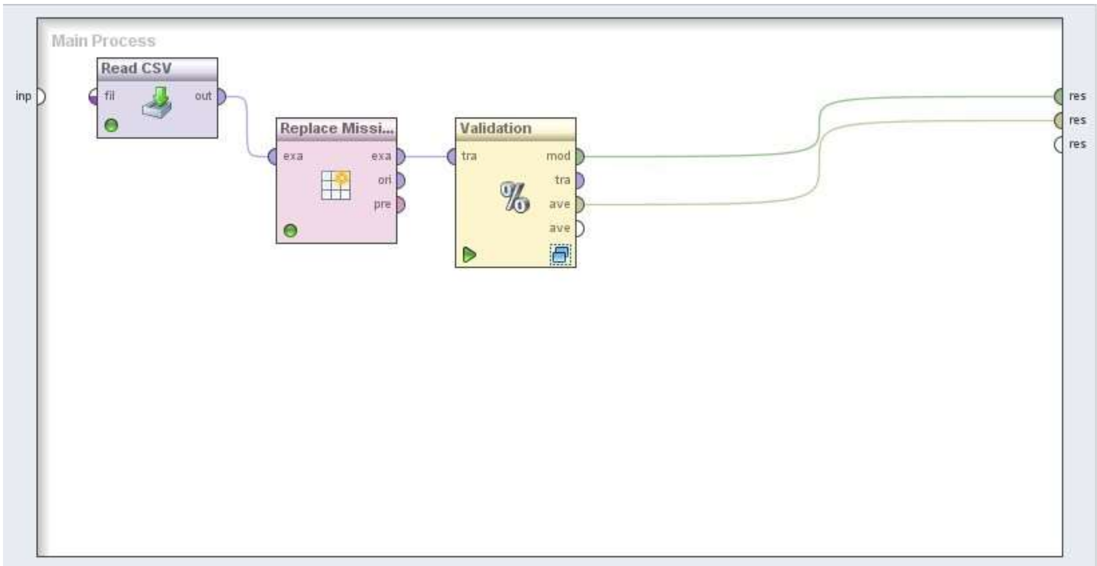

Figure 4: Modeling SVM Algorithm on the Outputs of PSO-1 Algorithm

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

In the figure above, the training data relating to the years 2011 and 2014 are entered into the model. Then outliers were removed of data and finally output was entered the Validation control. Validation controls by double-clicking on a window in the window in accordance with the following two sections observed. Double-clicking on Validation control, a window has been opened and two parts can be seen in this window according to the following figure.

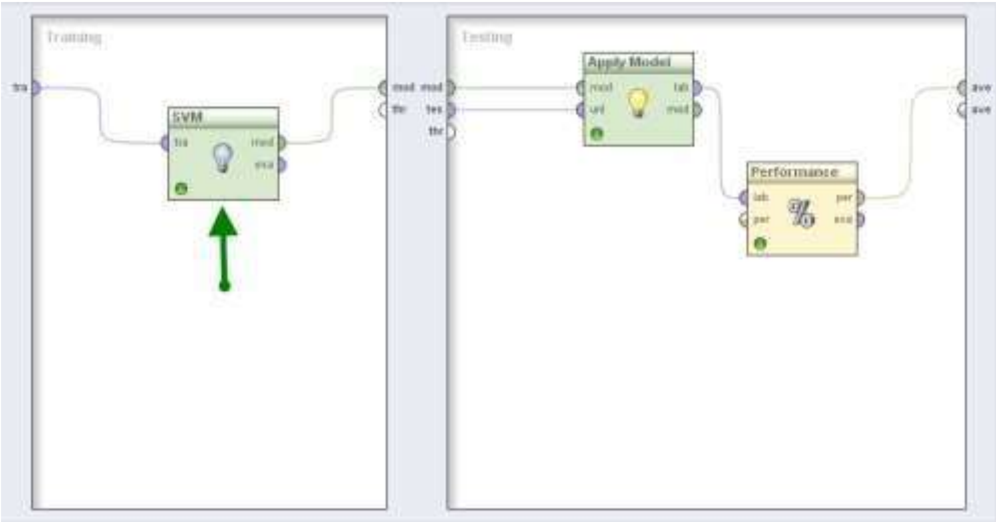

Figure 5: Modeling SVM Algorithm on the Outputs of PSO-2 Algorithm

As seen in the figure above, the model of SVM algorithm is in the Training section, which uses part of the training data, and training data is applied in the Testing section. In Testing, first the data for 2015, which is expected prediction to be make on it, is entered the model by SVM.

Apply Model Control is made by SVM algorithm. This model has input and output ports which input ports is related to the experimental data and SVM algorithm and output ports are used for predictions made in accordance with the experimental data. Output of Model Apply Control to evaluate the accuracy, errors and other measures is entered to Performance control.

Finally, accuracy, errors and other necessary criteria is evaluated. Finally, after the implementation of the proposed method, the accuracy of the model PSO_SVM is as follows.

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

PerformanceVector:

accuracy: 98.63% +/- 0.19% (mikro: 98.63%)

ConfusionMatrix:

True:	3	6	7	8	2	5	0	1
3:	1664	10	3	0	0	0	0	0
6:	31	1986	0	8	0	0	0	0
7:	0	0	1655	10	2	0	0	0
8:	0	8	11	2110	1	13	0	0
2:	0	0	1	0	1431	7	0	3
5:	0	0	0	0	12	2030	12	4
0:	0	0	0	0	0	12	1157	17
1:	0	0	0	0	4	2	8	810

classification_error: 1.37% +/- 0.19% (mikro: 1.37%)

ConfusionMatrix:

True:	3	6	7	8	2	5	0	1
3:	1664	10	3	0	0	0	0	0
6:	31	1986	0	8	0	0	0	0
7:	0	0	1655	10	2	0	0	0
8:	0	8	11	2110	1	13	0	0
2:	0	0	1	0	1431	7	0	3
5:	0	0	0	0	12	2030	12	4
0:	0	0	0	0	0	12	1157	17
1:	0	0	0	0	4	2	8	810

absolute_error: 0.017 +/- 0.005 (mikro: 0.017 +/- 0.207)

root_mean_squared_error: 0.198 +/- 0.062 (mikro: 0.208 +/- 0.000)

Figure 6: Accuracy, Error and the Average of Square Error of the Proposed PSO-SVM Method

As is clear from the above, the accuracy of the proposed method is equal to 98.63%, the error rate of the proposed method is also equal to 1.37%, RMSE is equal to 0.198.

Analysis and Evaluation:

In this section, the results of the proposed method will be evaluated and compared with other methods raised by other studies. After applying the proposed model, as well as implementation of other popular methods, results and performance of each will be examined.

In the following figure, the results of the implementation of KNN algorithm on the same dataset with the proposed method are shown.

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

PerformanceVector:
accuracy: 98.26% +/- 0.27% (mikro: 98.26%)
ConfusionMatrix:

True:	3	6	7	8	2	5	0	1
3:	1655	17	3	0	0	0	0	0
6:	34	1983	0	5	0	0	0	0
7:	6	0	1650	25	1	0	0	0
8:	0	4	15	2091	0	7	0	0
2:	0	0	2	1	1429	23	0	3
5:	0	0	0	6	15	2018	15	4
0:	0	0	0	0	0	16	1156	13
1:	0	0	0	0	5	0	6	814

classification_error: 1.74% +/- 0.27% (mikro: 1.74%)
ConfusionMatrix:

True:	3	6	7	8	2	5	0	1
3:	1655	17	3	0	0	0	0	0
6:	34	1983	0	5	0	0	0	0
7:	6	0	1650	25	1	0	0	0
8:	0	4	15	2091	0	7	0	0
2:	0	0	2	1	1429	23	0	3
5:	0	0	0	6	15	2018	15	4
0:	0	0	0	0	0	16	1156	13
1:	0	0	0	0	5	0	6	814

absolute_error: 0.017 +/- 0.003 (mikro: 0.017 +/- 0.131)
root_mean_squared_error: 0.131 +/- 0.010 (mikro: 0.132 +/- 0.000)

Figure 7: Summary of the KNN Algorithm Performance on Similar Data Set with Proposed Method

As is clear from the above, the accuracy of KNN algorithm on the same dataset with the proposed method is equal to 98.26%, the error rate of the proposed method is also equal to 1.74% and RMSE is equal to 0.131.

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

PerformanceVector:
accuracy: 96.00% +/- 0.88% (mikro: 96.00%)
ConfusionMatrix:

True:	3	6	7	8	2	5	0	1
3:	1627	3	4	0	0	0	0	0
6:	56	1990	0	64	0	0	0	0
7:	0	0	1610	42	17	0	0	0
8:	11	11	52	2002	2	29	0	0
2:	0	0	4	1	1395	16	0	4
5:	1	0	0	19	29	1935	6	3
0:	0	0	0	0	0	84	1168	53
1:	0	0	0	0	7	0	3	774

classification_error: 4.00% +/- 0.88% (mikro: 4.00%)
ConfusionMatrix:

True:	3	6	7	8	2	5	0	1
3:	1627	3	4	0	0	0	0	0
6:	56	1990	0	64	0	0	0	0
7:	0	0	1610	42	17	0	0	0
8:	11	11	52	2002	2	29	0	0
2:	0	0	4	1	1395	16	0	4
5:	1	0	0	19	29	1935	6	3
0:	0	0	0	0	0	84	1168	53
1:	0	0	0	0	7	0	3	774

absolute_error: 0.097 +/- 0.007 (mikro: 0.097 +/- 0.174)
root_mean_squared_error: 0.199 +/- 0.014 (mikro: 0.199 +/- 0.000)

Figure 8: Summary of the Naive Bayes Algorithm Performance on Similar Data Set with Proposed Method

As is clear from the above, the accuracy of the naive bayes algorithm on the same dataset with the proposed method is equal to 96.00%, the error rate of the proposed method is also equal to 4.00% and RMSE is equal to 0.1999.

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

```
PerformanceVector:
accuracy: 57.70% +/- 6.98% (mikro: 57.74%)
ConfusionMatrix:
True:   3       6       7       8       2       5       0       1
3:      419     1       3       0       0       0       0       0
6:      1234    1959    4       0       0       0       0       0
7:      0       0       396     0       99      0       0       70
8:      0       0       1297    2110    109     535     258     1
2:      0       0       0       0       193     0       0       0
5:      0       0       0       2       1054    1421    5       0
0:      0       0       0       0       0       17      884     737
1:      0       0       0       0       0       0       0       32
classification_error: 42.30% +/- 6.98% (mikro: 42.26%)
ConfusionMatrix:
True:   3       6       7       8       2       5       0       1
3:      419     1       3       0       0       0       0       0
6:      1234    1959    4       0       0       0       0       0
7:      0       0       396     0       99      0       0       70
8:      0       0       1297    2110    109     535     258     1
2:      0       0       0       0       193     0       0       0
5:      0       0       0       2       1054    1421    5       0
0:      0       0       0       0       0       17      884     737
1:      0       0       0       0       0       0       0       32
absolute_error: 0.454 +/- 0.075 (mikro: 0.454 +/- 0.332)
root_mean_squared_error: 0.559 +/- 0.066 (mikro: 0.563 +/- 0.000)
```

Figure 9: Summary of the Random Forest Algorithm Performance on Similar Data Set with Proposed Method

As is clear from the above, the accuracy of the Random Forest algorithm on the same dataset with the proposed method is equal to 57.70%, the error rate of the proposed method is also equal to 42.30% and RMSE is equal to 0.559.

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

```
PerformanceVector:
accuracy: 71.63% +/- 1.97% (mikro: 71.62%)
ConfusionMatrix:
True:   3       6       7       8       2       5       0       1
3:      1416    0       211     0       0       0       0       0
6:      248     1957    2       138     0       0       0       0
7:      31      0       1232    1       190     0       0       56
8:      0       47      62      1460    2       326     39      0
2:      0       0       150     0       1137    0       0       557
5:      0       0       13      529     121     1738    826     54
0:      0       0       0       0       0       0       312     92
1:      0       0       0       0       0       0       0       75
classification_error: 28.37% +/- 1.97% (mikro: 28.38%)
ConfusionMatrix:
True:   3       6       7       8       2       5       0       1
3:      1416    0       211     0       0       0       0       0
6:      248     1957    2       138     0       0       0       0
7:      31      0       1232    1       190     0       0       56
8:      0       47      62      1460    2       326     39      0
2:      0       0       150     0       1137    0       0       557
5:      0       0       13      529     121     1738    826     54
0:      0       0       0       0       0       0       312     92
1:      0       0       0       0       0       0       0       75
absolute_error: 0.444 +/- 0.030 (mikro: 0.444 +/- 0.735)
root_mean_squared_error: 0.858 +/- 0.030 (mikro: 0.859 +/- 0.000)
```

Figure 10: Summary of the Linear Regression Algorithm Performance on Similar Data Set with Proposed Method

As is clear from the above, the final precision of linear regression algorithm on the same dataset with the proposed method is equal to 71.63%, the error rate of the proposed method is also equal to 28.37% and RMSE is equal to 0.858.

Compare the Proposed Method with Other Methods:

In this study, a measure of accuracy that is one of the criteria in link predictionwas compared with others methods. In the table below, the figures the accuracy of the proposed method is shown with other popular methods. mLR, mSVM, CIT criteria are derived from [7].

Table 2: Comparison of Prediction Accuracy of the Proposed Method with other Methods (Accuracy Criteria)

Proposed Method	KNN	Naive Bayes	Random Forest	Linear regression	mLR	mSVM	CIT
98.63	98.3	96	57.7	71.63	98.35	98.35	98.34

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

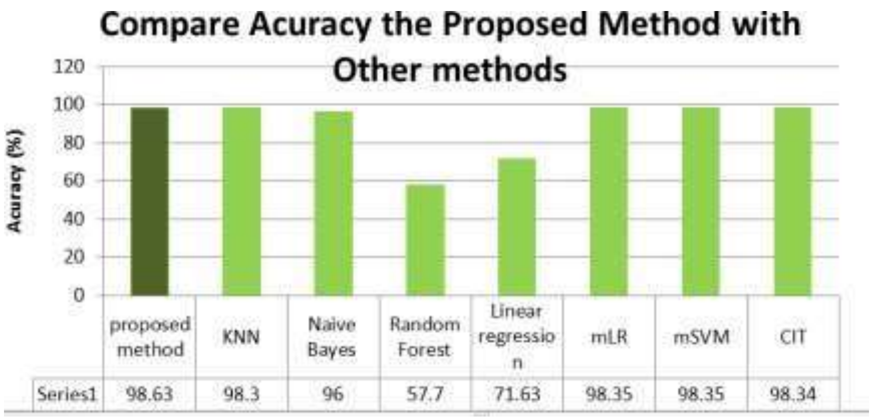

Figure 11: Comparison of the Prediction Accuracy of the Proposed Method with Other Methods (Accuracy Criteria)

As is clear from the figure above, the accuracy of the proposed method has improved compared with the accuracy of the individual methods. To calculate the average improvement achieved with the methods individually, the percentage of improving the method individually and mean has been calculated. The average percent improvement in the accuracy of the proposed method compared to other methods is about 16%.

Error criterion, one of the most important criteria in link prediction will be compared with other methods. In the table below the figures the error rate of the proposed method is shown compared to other popular methods.

Table 3: Comparison of the Prediction Error of the Proposed Method with Other Methods

Proposed Method	KNN	Naive Bayes	Random Forest	Linear regression	mLR	mSVM	CIT
1.37	1.74	4	42.3	28.37	1.65	1.65	1.66

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

International Academic Journal of Science and Engineering,
Vol. 5, No. 1, pp. 222-240.

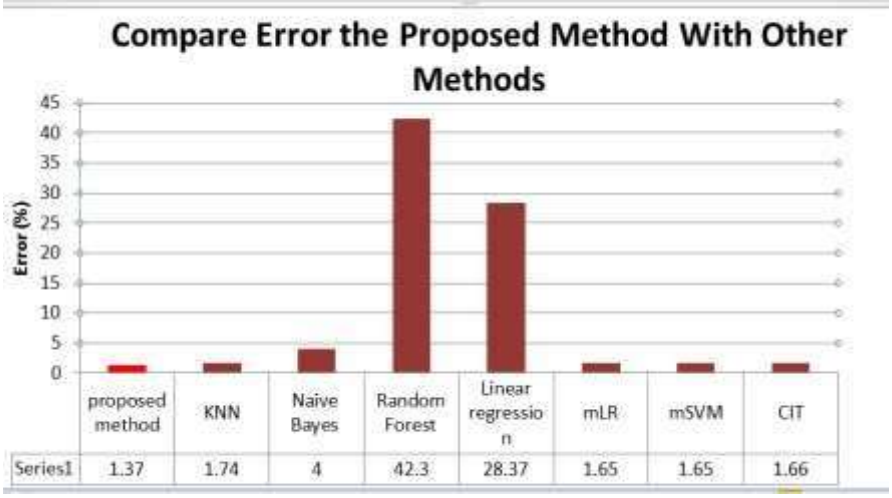

Figure 12: Comparison of the Prediction Error of the Proposed Method with Other Methods (Error Rate Criteria)

As is clear from the above, the error rate of the proposed method has been improved compared with the accuracy of the individual methods. Therefore, average percentage of improvement in the accuracy of the proposed method compared to other methods is about 47%.
RMSE criterion, which is one of the most used criterion in the link prediction methods are compared with other methods. In the following table, the numbers and figures related to RMSE of the proposed method compared to other popular methods is shown.

Table4: Comparison of the Prediction RMSE of the Proposed Method with Other Methods (RMSE Criteria)

Proposed Method	KNN	Naive Bayes	Random Forest	Linear regression	mLR	mSVM	CIT
0.198	0.131	0.199	0.559	0.858	0.275	0.219	0.193

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

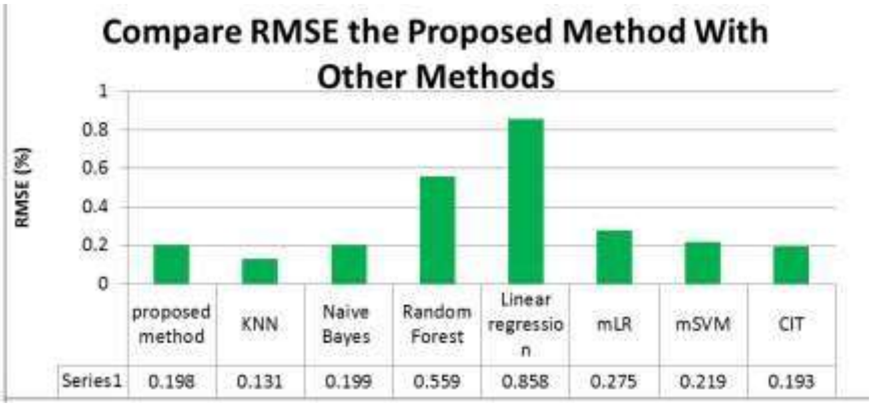

Figure 13: Comparison of the Prediction RMSE of the Proposed Method with Other Methods (RMSE Criteria)

Finally, the mean percentage of improvement RMSE of the proposed method compared to other methods is about 32%.

Conclusion:

In this study, the basic concepts of the study have been studied. The areas of dynamic co-authorship networks were reviewed and link prediction on this type of social networks have been investigated. The regulatory procedures and classification models were used to predict link. In this study, the support vector machine learning algorithm was used to train and test the models, and applied on the dataset.
Obtained results were compared with other similar methods and algorithms, and we found that the proposed method has the accuracy of 98.63, and compared to the studied methods such as KNN or Random Forest has least amount of error in future links prediction, ie 1.37. In future research programs, we will try to use other feature selection methods, such as genetic and bee algorithms to select outstanding features to increase accuracy, and compare the results obtained with the proposed method in this article. Also, the use of other algorithms, such as decision tree algorithm, neural networks, artificial neural networks, algorithms similar to it, and then comparing the results obtained with the proposed method, are among issues which will be considered in the future research.

References:

1. M. Nikzad, H. R. Jamali, N. Hariri (2011). Patterns of Iranian Co-Authorship Networks in Social Sciences: A Comparative Study, Library & Information Science Research, 33, 313–319
2. M. E. J. Newman (2000). Who is the Best Connected Scientist? A Study of Scientific Co-Authorship Network, physics.soc-ph arxiv.org,
3. M. A.Hasan, M. J. Zaki (2011). Social Network Data Analytics, Computer Science Springer, 243-275
4. D. L. Nowell, J. Kleinberg (2003) .The Link-Prediction Problem for Social Networks, conference on Information and knowledge management, 556-559
5. R. Cheng, Y. Jin (2015). A social learning particle swarm optimization algorithm for scalable optimization, Information Sciences, 291, 43-60

Received:12 February 2018/Accepted:16 May/Published:1 June 2018

6. P. Arora, Deepali, S. Varshney (2016). Analysis of K-Means and K-Medoids Algorithm For Big Data, Procedia Computer Science, 78, 507-512
7. N. Pobiedina, R. Ichise (2016). Citation count prediction as a link prediction problem, Applied Intelligence, 44, 252-268
8. Y. Wang, Q. Wang, H. Zhang, K. An, X.Ye, Y. Sun (2016). Improved Particle Swarm Optimization Algorithm for Optimization of Power Communication Network, International Journal of Grid and Distributed Computing, 9, 225-236
9. V. K. Dehariya, S. K. Shrivastava, R.C. Jain (2010). Clustering of Image Data Set Using K-Means and Fuzzy K-Means Algorithms, International Conference on Computational Intelligence and Communication Networks (CICN) IEEE, 386-391

Received:12 February 2018/Accepted:16 May/Published:1 June 2018