https://doi.org/10.31436/iiumej.v22i1.1622

PREDICTING TRUST IN A SOCIAL NETWORK BASED ON STRUCTURAL SIMILARITIES USING A MULTI-LAYERED PERCEPTRON NEURAL NETWORK

AMIR HOSSEIN DANESH[1] AND HOSSEIN SHIRGAHI[2*]

[1]Sogang University, Seoul, South Korea
[2]Computer Engineering Department, Islamic Azad University, Jouybar Branch, Jouybar, Iran

[*]Corresponding author: hossein.shirgahi@gmail.com

(Received: 9th September 2020; Accepted: 14th October 2020; Published on-line: 4th January 2021)

ABSTRACT: Although research on social networks is progressing rapidly, the positive and negative effects of this area should be evaluated. One of the problems is that social networks are very broad and anyone can have influence on them. This matter can cause the issue of people with different beliefs. Therefore, determining the amount of trust to various resources on social networks, and especially resources for which there is no previous history on the web, is one of the main challenges in this field. In this paper, we present a method for predicting trust in a social network by structural similarities through the neural network. In this method, the web of trust data set is converted to a structural similarity data set based on the similarity of the trustors and trustees first. Then, on the created data set, a part of the data set is considered as the training data and it is trained based on the multilayer perceptron neural network and then the trained neural network is tested based on the test data. In the proposed method, the MSE value is less than 0.01, which has improved more than 0.02 compared to previous methods. Based on the obtained results, the proposed method has provided acceptable accuracy.

ABSTRAK: Walaupun kajian tentang rangkaian sosial adalah sangat pesat, kesan positif dan negatif dalam ruang lingkup ini perlu dinilai. Masalah rangkaian sosial adalah sangat luas dan sesiapa sahaja boleh terpengaruh. Perkara ini akan menyebabkan manusia dengan pelbagai isu kepercayaan. Oleh itu, menentukan nilai kepercayaan melalui pelbagai sumber dalam rangkaian sosial, terutama sumber-sumber yang tidak mempunyai sejarah lepas dalam web, adalah salah satu cabaran dalam bidang ini. Kajian ini membentangkan jangkaan kepercayaan dalam rangkaian sosial melalui persamaan struktur dengan menggunakan rangkaian neural. Kaedah ini ditentukan dengan menukar set data web kepercayaan kepada struktur set data hampir sama berdasarkan kesamaan pemegang dan pemberi amanah. Kemudian, sebilangan set data yang telah dibina ini dipertimbangkan sebagai data latihan dan ia dilatih berdasarkan rangkaian neural perseptron berbagai lapisan dan kemudian rangkaian neural yang terlatih ini diuji berdasarkan data ujian. Dalam kaedah yang dicadangkan ini, nilai MSE adalah kurang daripada 0.01, di mana telah diperbaiki kepada 0.02 lebih daripada kaedah-kaedah sebelum ini. Berdasarkan dapatan kajian, didapati kaedah yang dicadangkan ini menunjukkan ketepatan yang boleh diterima.

KEYWORDS: *social network; trust; structural similarity; web of trust; neural network*

1. INTRODUCTION

In recent years, topics such as social networks and semantic web have become the main topics of research in emerging social media. The philosophy of creating social networks is such as world wide web, in which everyone can produce information or use the information of others. So far, most of the researches that has been done in the field of semantic web and social network platform has been related to determine the standards of communication rules and facts such as XML, RDF, RDF diagram, OWL, etc., which create the necessary basis for building a social network.

It is clear we can't expect every user to know how reliable each resource is. Also, due to the objective nature of the trust, the overall values of believability cannot be determined. It must be determined to what extent each source of information can be trusted. One solution is to keep all the information high quality and with no contradicted on social network. But due to the intensity and variety of resources, this is almost impossible. As a result, assuming the variety of information quality, an efficient solution must be provided that can be considered effective.

The researchers of WWW found that a method to do this is using quality assertion in the link structure between pages [1]. Such a function is also used in social networks. Often in order to evaluate trust in social networks, web of trust is made based on users and their interactions and analysis of the above web of trust is done. The created web of trust may be used repeatedly to determine a user's trust to the other users. Remember that unlike site ranking, the result of calculations on web of trust networks is not an accumulation of each user's trust ability. Instead, each user receives a set of his personal trusts that its amount is vary greatly from person to person. Another challenge in this area is the type of assertion to the content of social networks, which can be logical or probable. If the user assertion is a logical type, each user can have agree or disagree assertion with each content. If all assertions are inconsistent about one content, that content can be believed with certainty and the assertions can be combined logically. Otherwise, probabilistic calculations can be used. Of course, the purpose is not to understand the reasons of the assertions. Rather, due to the difficulty of establishing a degree of belief in an assertion that is explicitly stated by one or more resources on the social network, the goal is finding a solution for the above problem.

The basic model is that a user's belief in an assertion should be a function of the trust to the source that provides it. If the user's belief in the source is clear, the user's belief in the person who determines the amount of trust for the source can be calculated. This is done in a variety of ways that is related to how people's beliefs are formed. If there are trusts from A to B scores at U amount and from B to C scores at V amount, then A will have a ratio of trust to C scores at T amount, which T is itself a function of U and V.

This rule can be used to measure user trust in resources and also to answer the question of how much each user can trust sources that he or she does not know directly. This law is based on the hierarchical publication of trust. Admissible methods of combining trusts that allow accurate and local calculation of derived trusts have limitations. These limitations in the composition of beliefs also cause to do only with the use of local information [2].

Consider a system consisting of n users that have m assertions in overall. Since each assertion is considered separately, we introduce the system in such a way that there seems to be only one assertion. Beliefs: Each user can affect his personal belief in assertion that ranges from 0 to 1. The high numerical value of this range indicates that the assertion is accurate and reliable. Suppose B_i represents the personal belief of the user i in the assertion.

If the user *i* does not have belief, we set B_i to zero. The set of personal beliefs in the assertion is shown as a B-column vector.
Trusts: User **i** can trust user **j** as $\mathbf{T_{ij}}$. It ranges from 0 to 1. The high numerical value of $\mathbf{T_{ij}}$ shows that user **i** trusts user j or they have the same interests. If the trust value is not specified, we set $\mathbf{T_{ij}}$ to zero. Note that $\mathbf{T_{ij}}$ and $\mathbf{T_{ji}}$ do not have to be equal.

The set of personal trusts can be represented as a matrix **T** with **n** * **n** dimensions. $\mathbf{T_i}$ represents the linear vector of the user *i*'s personal trust to the other users [3].

Integration: Web of Trust creates a structure in which we can calculate the level of belief of each user of the assertion. To distinguish it from the personal beliefs of the user (B), we call it integration beliefs.

In this article, we do not deal with the details of users' beliefs and individual assertions on the content; rather, by using the web of trust and the specific values of trust in the data set, we predict the trust for nodes that do not have a direct edge. We perform predicting trust based on different structural similarities and their combinations using a multilayer perceptron neural network in social network. In previous research, a structural similarity method is usually used. As another innovation in this article, we used the similarity of both trustor and trustee to predict trust, but in previous methods, one of these two similarities is often used.

2. PREVIOUS WORKS REVIEW

In general, there are two models of trust based on web of trust data set: the binary trust model with the probability distribution available in [4,5] and the multilevel trust model without the probability distribution [6]. Most e-commerce websites, such as eBay, Ali baba, Amazon and Yelp, have adopted a multi-level trust model. In some studies, the display of trust has been generalized in statistical distribution. In particular, reference [7] suggests a subjective logic to express distrust and to evaluate the trust probability distribution. But this method can only be used for the binary trust model. A variety of methods have been proposed to evaluate or trust propagation in different types of networks [8] and [9]. For example, in reference [9], a scoring system called Eigen Trust is created based on peers' historical functionality for peer to peer networks. However, this method is not directly applicable to assess the trust of social networks, because two people may have disparate opinions on a social network about the reliability of a person or party.

According to reference [10], for each person, a global reputation on the network is evaluated from the view of valid nodes. A Tidal Trust algorithm was proposed in reference [6], that infer and speculate trust relations using continuous scale ranking. This method collects trust data from all reference paths according to the shortest path from one source to the destination . It then uses the weighted average method to calculate the total trust value. A trust inferential model called Sunny is proposed in reference [4] to measure trust through possible sampling and to reduce the impact of longer-distance routes (usually for low-confidence estimates). A flow-based trust assessment is proposed that considers the dependence of the path using network flows, and the trust models reduce or spread trust using expanding the flow associated with each node in the reference [5].

Reference [11] uses a matrix to display direct trust relationships between users on a social network and uses the Breadth-First search algorithm to enhance users trust. A binary decision chart based method [13] is proposed to evaluate the trust between the two parties, but it is only applicable for the binary trust model in the reference [12]. In this paper, the

relationship of trust between two directly connected parts is modeled using a multifaceted trust model with probability distribution. Then, a probabilistic hybrid method based on multi-valued decision diagram (MDD)'s is presented to determine the probability distribution of different levels of trust between the two parties on the social network. The parties can be connected through multiple routes, including a direct link or several indirect links. Dependence between different paths during the creation of the MDD model has been investigated. In reference [14], the TILLIT method is proposed, which is a pattern based on a combination of trust inference and user similarity. This similarity is based on the structure of the trust diagram and user's trust behavior. This method is unlike other filter-based approaches that have used user rankings.

The trust assessment is performed based on the reputation parameters of the recommendation according to the users' profiles in the reference [3]. Its purpose is to provide a fuzzy system for evaluating trust based on users' profiles, examining the accuracy of the fuzzy system for evaluating trust, and examining the time complexity of the fuzzy system for evaluating trust in the Semantic Web. In this research, user profile data set based on semantic parameters has been used. For this purpose, they have considered their desired data set from CTI Depaul site, which includes a list of pages, user sessions, pages visited in each user session, and the time users visited the pages.
mirroring trust is used to predict trust in reference [2], which estimates the amount of trust based on the degree of similarity between the trustor and the trustee. In this paper, the parameters degree, degree quality and local density in social networks have been used to determine similarity criteria and in order to calculate the similarity of each of these parameters, a fuzzy system is used.

Shirgahi et al. [1] gained the assessment of trust in the Semantic Web and Social Network by clustering the Web of trust in their research first, after that they used distributed aggregation of indirect trust paths to predict trust for nodes that do not have direct trust. This method has an acceptable execution time.

3. MULTI-LAYERED PERCEPTRON NETWORK

Past research shows that among the various methods of the neural network, the multi-layer perceptron network (MLP) with the law of feed forward back propagation is one of the most widely used, basic, and at the same time simplest models of neural network available. Also, MLP has been evaluated a suitable method in the field of estimating unknown parameters.

The learning algorithm in these types of networks is a type of error correction algorithm, which is called error Back Propagation algorithm or back Propagation briefly. These types of networks have an input layer, one or more hidden layers, and one output layer, and the information moves only in one direction, the direction of which is forward. In fact, the information starts from the input nodes and passes through the hidden layer (s) to the output nodes. So, there is no feedback. This means that the output of each layer only affects the next layer and does not change its own one (Fig. 1). Each network cell has a nonlinear function at the output and is derived for all inputs. The structure of the neural network used in this study is the back Propagation neural network with two hidden layers and one output layer. Also, the artificial neural network model is trained using the train set and different combinations of inputs are created. Therefore, to achieve these goals, 60% of the data are used for training, 20% for testing and 20% for evaluation.

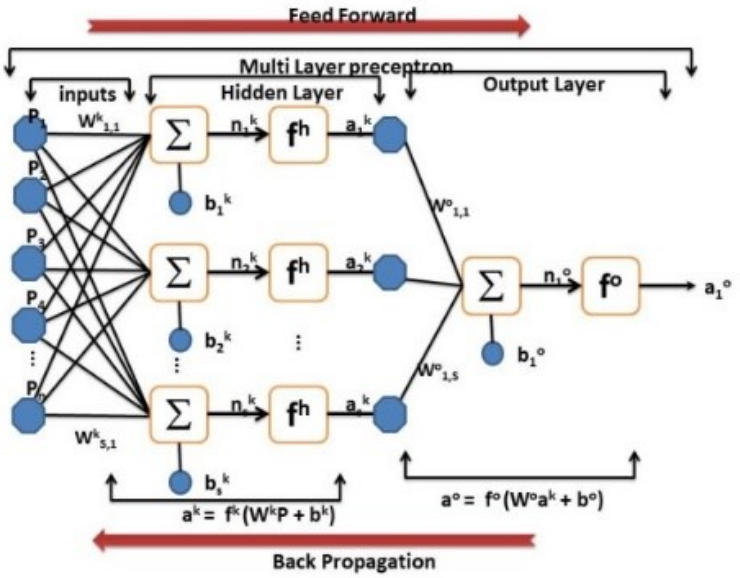


Fig. 1: View of the two-layer artificial neural network back propagation.

4. PROPOSED METHOD

In this section, the various criteria for structural similarity in the social network will be described first, then the structure of converting the social network data set to the structural similarity data set will be shown, and finally the structure of the proposed multi-layered perceptron neural network will be explained.

4.1 Structural Similarity Criteria in Social Network

In this article, to determine the structural similarity in the social network, we have used a series of structural similarities of the weighted directional graph and a series of structural parameters in the social network. Before explaining the above criteria, we first describe the structure of web of trust.

A web of trust is a weighted directional graph as Fig. 1 and the values of the weights of the nodes and edges are in the range [0, 1]. The nodes in this part of the study are the trustors and trustees. The weight of each node determines the global reputation of each node which is determined by Eq. 1. The weight of each edge **u** to **v** determines the value of direct trust obtained by direct interactions from **u** to **v**.

$$\mathrm{Rep_i} = \begin{cases} 0 & \mathrm{N_T} = 0 \\ \dfrac{\sum_{j \neq i, T_{ji} \neq 0} T_{ji}}{\mathrm{N_T}} & \mathrm{N_T} \neq 0 \end{cases} \quad (1)$$

In Eq. 1, $\mathbf{T_{ji}}$ is the value of direct trust from **j**-node to **i**-node and **NT** is the total number of nodes that trust directly on **i**-node.

https://doi.org/10.31436/iiumej.v22i1.1622

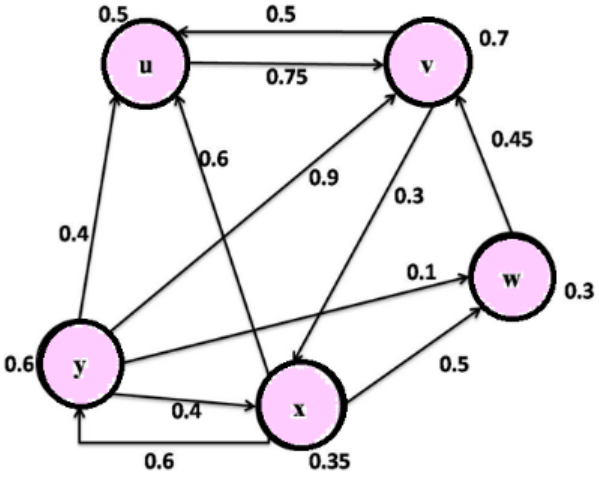


Fig. 2: An example of a web of trust network.

4.2 Graph Structural Similarities

In this paper, 5 structural similarities based on edges are used. Of course, the relationships of these similarities are weighed based on the amount of trust that exists between the nodes in the web of trust. Since the numerator of deduction of all these equations of edges based structural similarity, are according to the intersection of two **u** and **v** nodes, we define the weighted intersection of nodes according to Eq. 2.

In Eq. 2, **V** is a set of web of trust nodes and $|T_{ui} - T_{vi}|$ is the absolute value between $\mathbf{T_{ui}}$ and $\mathbf{T_{vi}}$.

$$|u \cap v| = \sum_{i \in V, T_{ui} \neq 0, T_{vi} \neq 0} 1 - |T_{ui} - T_{vi}| \qquad (2)$$

In this study, we weighted the structural similarities of Cosine, Dice, HDI, HPI, and Jaccard. These similarities are calculated by Eqs. (3) to (7), respectively. In these equations, $|u \cap v|$ is the weighted intersection of nodes u and v in accordance with Eq. 2 and the $\mathbf{R_{epu}}$ and $\mathbf{R_{epv}}$ parameters are the reputation values of nodes **u** and **v**, respectively, based on Eq. 1.

$$\mathrm{Sim}_{\mathrm{Cosine}}(u,v) = \frac{|u \cap v|}{\sqrt{\mathrm{Rep}_u} * \sqrt{\mathrm{Rep}_v}} \qquad (3)$$

$$\mathrm{Sim}_{\mathrm{Dice}}(u,v) = \frac{2 * |u \cap v|}{\mathrm{Rep}_u + \mathrm{Rep}_v} \qquad (4)$$

$$\mathrm{Sim}_{\mathrm{HDI}}(u,v) = \frac{|u \cap v|}{\max(\mathrm{Rep}_u, \mathrm{Rep}_v)} \qquad (5)$$

$$\mathrm{Sim}_{\mathrm{HPI}}(u,v) = \frac{|u \cap v|}{\min(\mathrm{Rep}_u, \mathrm{Rep}_v)} \qquad (6)$$

$$\mathrm{Sim}_{\mathrm{Jaccard}}(u,v) = \frac{|u \cap v|}{\mathrm{Rep}_u + \mathrm{Rep}_v - |u \cap v|} \qquad (7)$$

4.3 Structural Parameters in the Social Network

In web of trust networks that are used to assess trust in social networks or semantic web, different parameters are considered to compare networks and the similarity of nodes. Some of these parameters are semantic and are often used in semantic web, and others are structural parameters that are often considered in social networks to show similarities. In this part of the research, we describe some of the most important parameters of social networks that are used for the structural similarities of nodes.

Out degree:

The output degree parameter is used to determine how much a node exchanges information with other network nodes in a network. For discrete networks, the normal value of the out degree is obtained by dividing the number of output edges of a node by the maximum number of possible edges for that node (which can be the graph degree). For example, if there are five nodes in a network and one source node connects with three of the other four nodes through an edge, the out degree of the source node is 0.75.

In order to extend the out degree for weighted networks with weighted nodes, the values of these edges are weighted based on the importance of the nodes in the other heads leading to these edges .In this method, a node that is connected to other more important nodes has a higher out degree rather than a node that is connected to other nodes of the same size or less importance. In fact, in web of trust, not only the number of outgoing edges is important, but also the value of these edges and the reputation value of the destination nodes in the edges. Because the value of the edges indicates the value of direct trust between the two nodes and the value (weight) of the destination nodes and the reputation value of the destination nodes. Based on the explanations given in this study, we used Eq. 1 to determine the out degree.

$$D_{out}(i) = \frac{1}{1-Rep_i} \sum_{j \neq i} Rep_j \times T_{ij} \quad (8)$$

In Eq. 8, in addition to the reputation of destination nodes and the direct trust from the source node to the destination node, the value of the source node reputation is also effective. In Eq. 8, $\mathbf{R_{epi}}$ is the source node reputation, $\mathbf{R_{epj}}$ is the destination nodes reputation, $\mathbf{T_{ij}}$ is the value of the direct trust from **i**-node to **j**-node, and $\mathbf{D_{out}}$ (i) is the out degree of **i**-node (source).

Since a node with a higher out degree can transmit more information to other nodes, so a node with a higher out degree can transmit faster or more appropriate information to other nodes compared with a node with a lower out degree. On the other hand, the out degree has no effect on the node that received the information. Therefore, we cannot expect the out degree to be effective on the time required for a node to receive information. The degree of input in many researches in web of trust is also called degree of reputation. By combining weighted nodes, the in degree can be defined based on the Eq. 9.

$$D_{in}(i) = \frac{1}{1-Rep_i} \sum_{j \neq i} Rep_j \times T_{ji} \quad (9)$$

In Eq. 9, $\mathbf{R_{epi}}$ is the source node reputation, $\mathbf{R_{epj}}$ is the destination nodes reputation, $\mathbf{T_{ij}}$ is the value of the direct trust from **j**-node to **i**-node, and $\mathbf{D_{in}}$ (i) is the in degree of **i**-node (source). The input and output degrees are the same for symmetric networks, but these degrees are usually different in web of trust.

In degree:

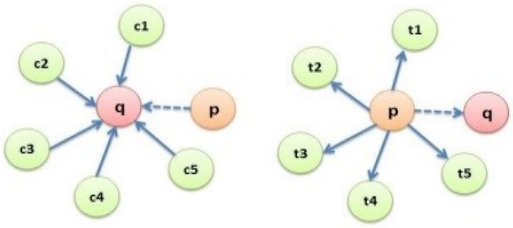
The input degree parameter is used to determine how much a node receives information from other nodes. For discrete networks, the normalized value of the in degree is obtained by dividing the number of input edges of a node by the maximum number of possible edges for that node (which can be the degree of the graph). The advogato social network data set is a directed graph with 51332 edges and 47300 nodes. Each record has 3 attributes, respectively the trustor user, the trustee user, and the value of trust. In this article, we have created a structural similarity data set using information about the edges and structural similarities of nodes. It has 51332 records, and each record is obtained based on the structural similarity analysis of each record in the advogato social network data set.

The structural similarity data set has 13 features for each $\mathbf{A_i}$ record from the advogato social network data set. $\mathbf{A_i},1$ is the number of a trustor node that trusts the trustee node $\mathbf{A_i},2$ as much as $\mathbf{A_i}$, 3 in the **i**-th record of the data set.

4.4 Converting Web of Trust Data Set to a Structural Similarity Data Set

In this study, converting the web of trust data set to a structural similarity dataset is done based on graph structural similarity and structural parameters in the social web. First, look at Fig. 3. The main problem is that the trustor **p**-node has no previous interaction with the trustee node **q**. The goal is to estimate the value of **p** to **q** based on the structural similarities between nodes **p** and **q** and other nodes. We can obtain the structural similarity of the **q**-node with all the trustee nodes with which the **p**-node has previously interacted (Fig. 3a). Based on this, the trust value can be predicted to determine the structural similarities of **p**-node with other nodes.

In addition, we can obtain the structural similarity of **p**-node with all other trustor nodes that have previously interacted with q-node (Fig. 3b) and predict the trust value on this basis. In this study, we used a combination of two structural similarities between trustors and trustees. Also, the out degree of the **p**-node and the in degree of the trustee nodes have a large effect on the prediction of the trust. As a result, we consider both of these features.



(a) (b)

Fig. 3. Trustees Similarities and Trustors Similarities in Predicting Trust
a) Trustors Similarities, b) Trustees Similarities.

The general structure of converting web of trust data set to a structural similarity data set is shown in Fig. 4. In the social network data set, we have an $\mathbf{A_{n*3}}$ matrix that n is the total number of edges and the three columns in **A**, respectively, are the number of trustor nodes, the number of trustee nodes, and the trust value corresponding to each trustor node to the trustee node. The structural similarity data set is a $\mathbf{B_{n*13}}$ matrix, which **n** is the total number of edges, columns 1 to 12 are the structural similarity features, and column 13 is the trust value of the corresponding record in matrix **A**. Here's how to calculate the above 13 features: the first 5 features are obtained based on the structural similarity of $\mathbf{A_{i,1}}$ and $\mathbf{A_{i,2}}$ nodes in terms of the similarity of the trustees to them, respectively, according to the similarities of Cosine, Dice, HDI, HPI and Jaccard and the sixth feature is obtained based on the normalized out degree of $\mathbf{A_{i,1}}$.

The features 7 to 11 are obtained based on the structural similarity of $\mathbf{A_{i,1}}$ and $\mathbf{A_{i,2}}$ nodes in terms of the similarity of the trustors to them, respectively, according to the similarities of Cosine, Dice, HDI, HPI and Jaccard. The twelfth attribute is based on the normalized value of $\mathbf{A_{i,2}}$ node's in degree and the thirteenth attribute indicates the trust value, in other words $\mathbf{A_{i,3}}$ node. Our goal of such a structural change has been to have a data set independence of the number of nodes, to have a trust value according to the different types of structural similarities between trustee and trustor nodes.

After creating the structural similarity data set, we randomly considered 70% of the data set as training data, 15% of the data as experimental data, and 15% of the data as validation data. You can see the general structure of the multilayer perceptron neural network in Fig. 5. In this structural similarity data set, the first 12 features are considered as input and the last feature is considered as output. Then, by teaching the above data set through the artificial neural network, the trust between structural similarity and trust value is obtained. Finally, the trust value for nodes that do not have a direct edge in the web of trust data set is estimated based on the trained neural network.

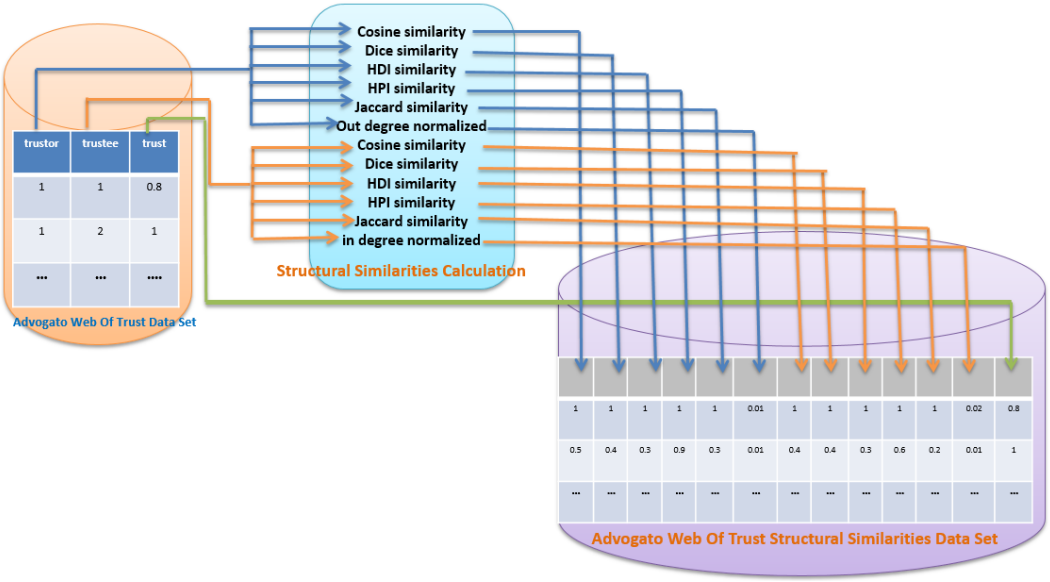


Fig. 4: Converting web of trust data set to a structural similarity data set.

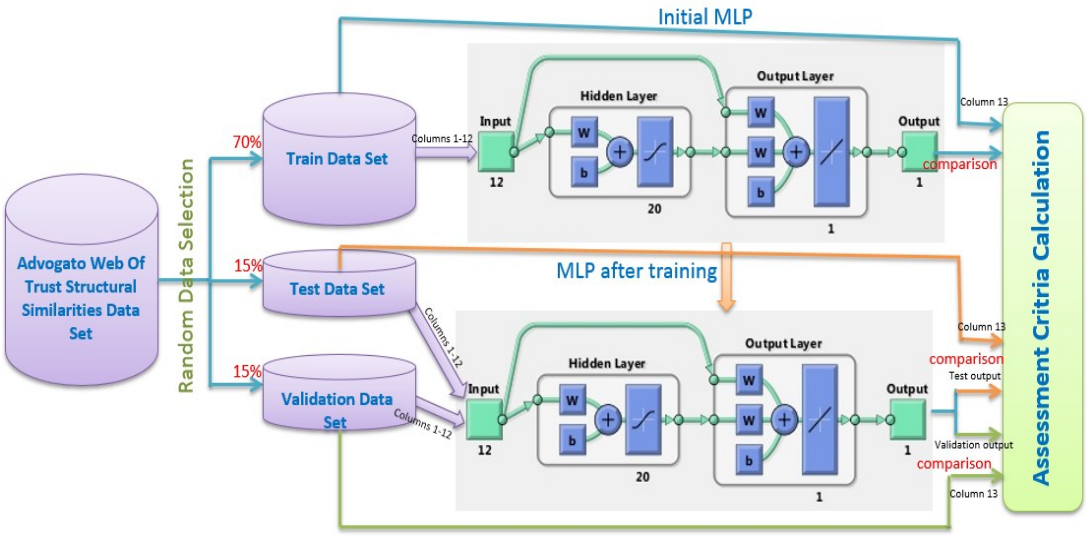


Fig. 5: General structure of multilayer perceptron neural network in the proposed method.

5. SIMULATIONS AND TESTS

The simulation of this article is done in MATLAB 2016. For this purpose, we used advogato web of trust data set, including a graph which characteristics are shown in Table 1 [15]. This data set has 51332 records; each record contains 3 columns (attributes) that

represent the trustor node, trustee node and trust value, respectively. In this data set, the trust values are positive and have one of the values 0.6, 0.8 and 1, and there are no distrust values. Clearly, if there is an edge between the two nodes, it shows that there has been a trusted interaction in the past. Therefore, trust is inferred from trustee behaviors, in other words, if the trustee's performance is appropriate, the trustor will trust it. Only if there is no direct edge between the two nodes, is it necessary to perform the trust prediction. In test operation, if there is a direct path between two nodes in web of trust, we also calculate the trust of the indirect path and compare the two values.

Table 1: Advogato web of trust data set characteristics

Characteristic	Value
Number of nodes	47300
Number of edges	51332
Graph density	0.0035
The most node degree	947
The least node degree	1

5.1 Evaluation Criteria

In this paper, the following evaluation criteria are used to evaluate the performance of the proposed method:

a) Mean squared error (MSE) and Root Mean squared error)RMSE): The mean square error of a set with n data is a method of estimating the error value, which is actually the difference between the estimated values and what is estimated.

$$\boldsymbol{MSE} = \frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}}(\boldsymbol{x_i} - \boldsymbol{y_i}\ )^{\mathbf{2}} \tag{10}$$

$$\boldsymbol{RMSE} = \sqrt{\frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}}(\boldsymbol{x_i} - \boldsymbol{y_i}\ )^{\mathbf{2}}} \tag{11}$$

Where $\mathbf{y_i}$ is the predicted value, $\mathbf{x_i}$ is the measured value, $\frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}}$ has done averaging operation and $\mathbf{x_i}$ calculates the value of square error of each data.

b) Scatter Index (SI): When the root mean square value is normalized to the measured average, it is sometimes referred to as the scatter index.

$$\boldsymbol{MSE} = \frac{\mathbf{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}}(\boldsymbol{x_i} - \boldsymbol{y_i}\ )^{\mathbf{2}} \tag{12}$$

c) Coefficient of Error (CE): It provides information on the accuracy of estimates of each measurement where CE_x shows the average estimation and **n** is the number of people.

$$\boldsymbol{CE_x} = \frac{\sqrt{\boldsymbol{CE_1}^{\mathbf{2}}+\boldsymbol{CE_2}^{\mathbf{2}}+\cdots+\boldsymbol{CE_n}^{\mathbf{2}}}}{\boldsymbol{n}} \tag{13}$$

d) Correlation Coefficient (R): This parameter expresses the power and direction of a linear correlation between two variables and its values are in the range -1 to 1. 1 indicates a One-to-one linearity and complete correlation, and -1 indicates a negative correlation. Coefficient of determination (R^2) describes the variance between two variables with a linear fit.

$$R = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad (14)$$

$$\boldsymbol{R^2} = \left[\frac{\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}\right]^2 \quad (15)$$

6. SIMULATION RESULTS

We use a multi-layered Perseptron neural network for training, test and validation based on structural similarities and the Levenberg training function. The maximum number of iterations is considered to be 1000 and the desired neural network is performed 10 times for 1 to 20 neurons in the hidden layer. We obtained the average results for each of the 1 to 20 neurons of hidden layer separately. You can see the implementation results in Fig. 6 to Fig. 11.

Figure 6 shows a comparison of the MSE criterion with different number of neurons. The lowest MSE is 0 with 3 neurons in training data, the lowest MSE is 0.00000014 with 4 neurons in test data, and the lowest MSE is 0.000012 with 1 neuron in validation data. Figure 7 shows a comparison of the RMSE criteria with different number of neurons. The lowest RMSE is 0 with 3 neurons in training data, the lowest RMSE is 0.00037 with 4 neurons in test data, and the lowest RMSE is 0.0035 with 1 neuron in validation data.

Figure 8 shows a comparison of the CE criteria with different number of neurons. The maximum CE is 1 with 3 neurons in training data, the maximum CE is 0.99982 with 4 neurons in test data, and the maximum CE is 0.99959 with 1 neuron in validation data. Figure 9 shows a comparison of the SI criteria with different number of neurons. The lowest SI is 0 with 3 neurons in training data, the lowest SI is 0.04527 with 4 neurons in test data, and the lowest SI is 0.42713 with 1 neuron in validation data.

Figure 10 shows a comparison of the R criteria with different number of neurons. The maximum R is 0.998225 with 3 neurons in training data, the maximum R is 0.98209 with 4 neurons in test data, and the maximum R is 0.97939 with 1 neuron in validation data. Figure 11 shows a comparison of run time with different number of neurons, the minimum run time is 10.452 seconds with 3 neurons. Figure 12 shows a comparison of the R criterion with 20 neurons. The R is 0.98225 in training data, the R is 0.98209 in test data, and the R is 0.97939 in validation data. In general, the criterion R is 0.97939.

Since the reference [14] researched in the field of predicting of trust on the Advogato data set, so we compared the obtained results with the RMSE criteria. You can see the comparison results in Table 2.

Table 2: Comparison of the proposed method with previous work

MSE	RMSE	Method
0.000012	0.0035	Structural similarity method with MLP
0.033708	0.1836	TILLIT [14]

https://doi.org/10.31436/iiumej.v22i1.1622

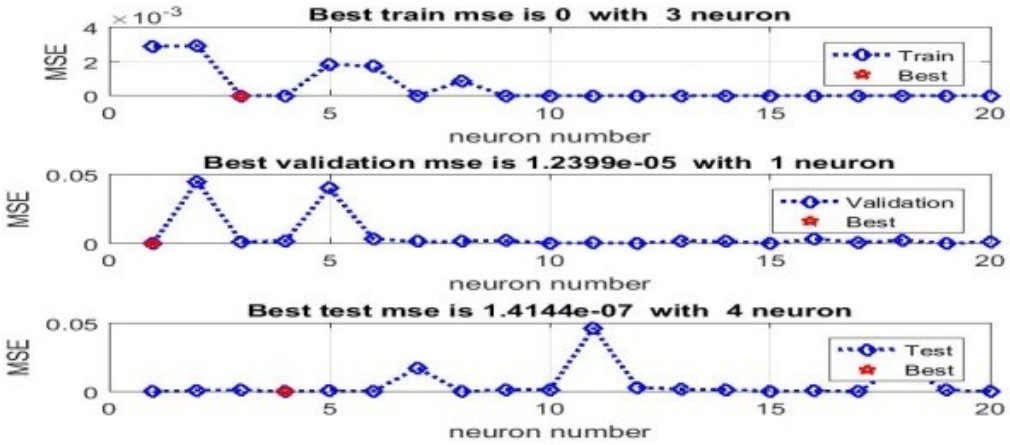


Fig. 6: Comparison of MSE criteria with a different number of neurons.

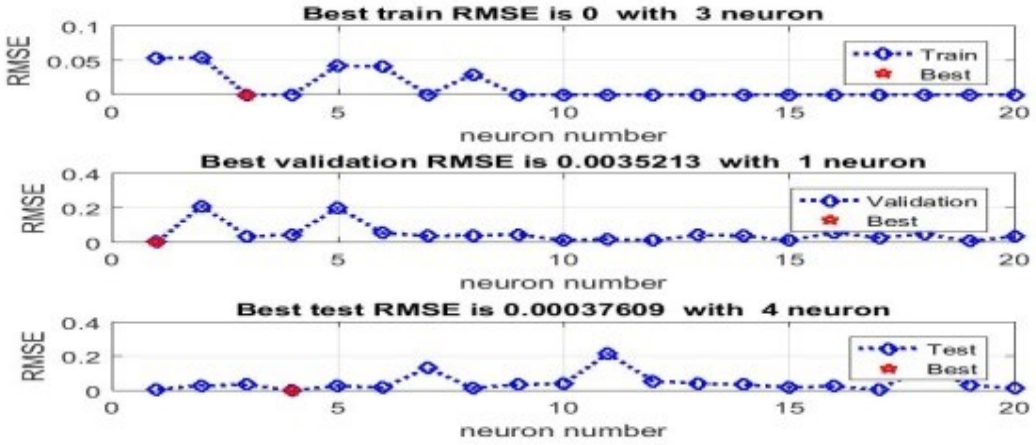


Fig. 7: Comparison of RMSE criteria with a different number of neurons.

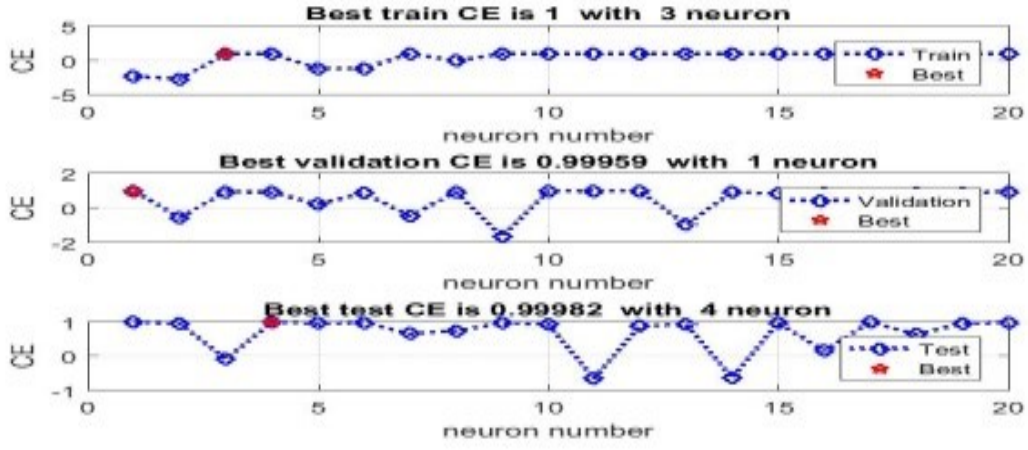


Fig. 8: Comparison of CE criteria with a different number of neurons.

https://doi.org/10.31436/iiumej.v22i1.1622

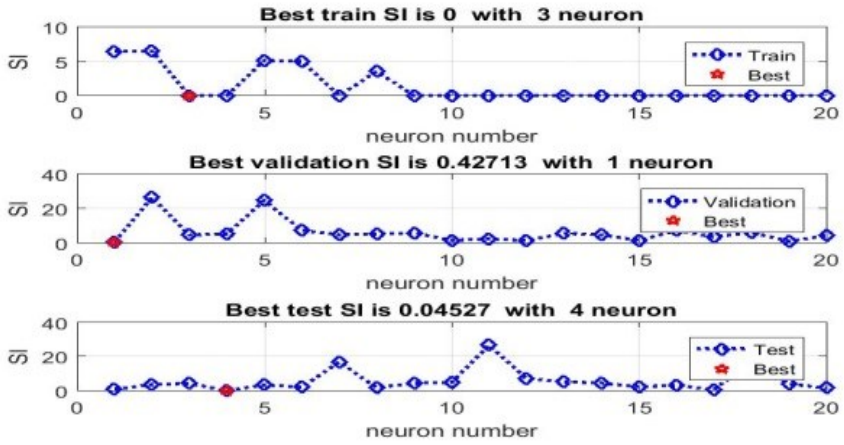


Fig. 9: Comparison of SI criteria with a different number of neurons.

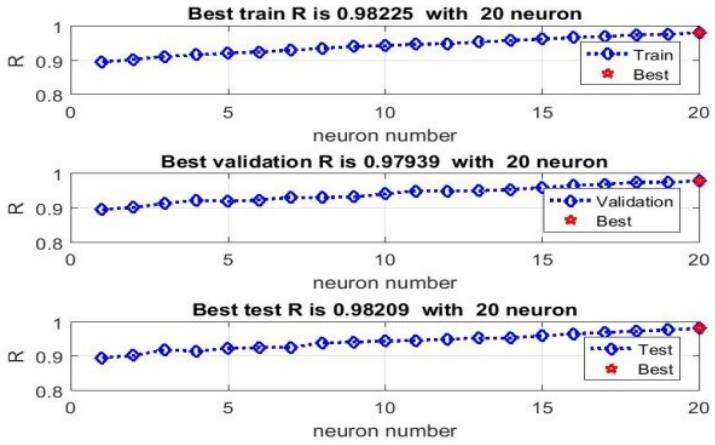


Fig. 10: Comparison of R criteria with a different number of neurons.

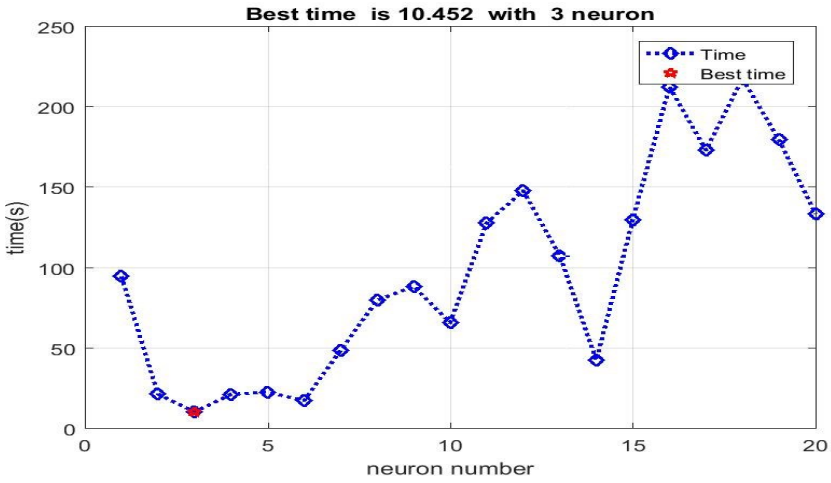


Fig. 11: Comparison of execution time with a different number of neurons.

https://doi.org/10.31436/iiumej.v22i1.1622

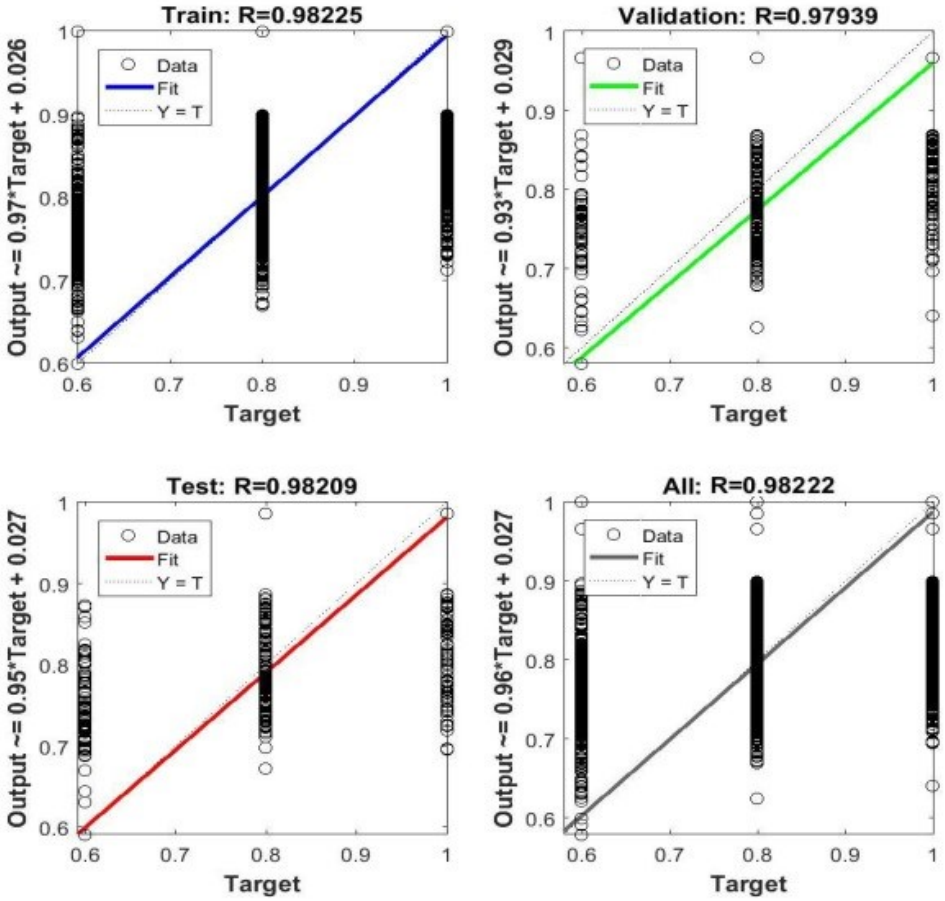


Fig. 12: Comparison of R criteria with 20 neurons in different states.

7. CONCLUSIONS AND SUGGESTIONS

In this research, after converting the web of trust data set into a structural similarity data set of trustors and trustees using a multilayer perceptron neural network, we trained the system. Then we implemented the test and validation data on the trained network. The simulation results are reported as follows:

For the validation data with 3 neurons in the hidden layer, the best MSE was 0.000012 and the best RMSE was 0.0035213. For the validation data with 1 neuron in the hidden layer, the most suitable CE was 0.99959 and the best SI was 0.42713. The best R for validation data with 20 neurons in the hidden layer has been obtained 0.97939. The proposed method has achieved good accuracy based on the results and it can be useful for predicting trust on social networks.

The proposed method is a centralized method, and on social networks with several hundred million users, predicting trust is a time consuming operation. Our suggestion for future work is that if each user identifies a set of trusted users (possibly a small set) accurately, a distributed method can be used that offers a less expensive, faster, and more scalable method. In this case, although it may be somewhat less accurate, such an approach would be more effective on the semantic web and neural network.

REFERENCES

[1] Shirgahi H, Mohsenzadeh M, Seyyed Javadi HH. (2017) Trust estimation of the semantic web using semantic web clustering. Journal of Experimental & Theoretical Artificial Intelligence, 29(3): 537-556.
[2] Shirgahi H, Mohsenzadeh M, Seyyed Javadi HH. (2018) A new method of trust mirroring estimation based on social networks parameters by fuzzy system. International Journal of Machine Learning and Cybernetics, 9(7): 1153-1168.
[3] Ashtari S, Danesh M, Shirgahi H. (2019) A novel user profile-based fuzzy approach for evaluating trust in semantic web. IIUM Engineering Journal, 20(1): 158-176.
[4] Kuter U, Golbeck J. (2007) A new algorithm for trust inference in social networks using probabilistic con_dence models. in Proc. AAAI, Vancouver, BC, Canada, pp. 1377-1382.
[5] Jiang W, Wu J, Li F, Wang G, Zheng H. (2016) Trust evaluation in online social networks using generalized network _ow. IEEE Trans. Comput., 65(3): 952-963.
[6] Golbeck J. (2005) Computing and applying trust inWeb-based social networks. Ph.D. dissertation, Dept. Comput. Sci., Univ. of Maryland, College Park, MD, USA.
[7] Jøsang A, Hayward R, Pope S. (2006) Trust network analysis with subjective logic. in Proc. ACSC, Hobart, TAS, Australia. pp. 85-94.
[8] Wang G, Jiang W, Wu J, Xiong Z. (2014) Fine-grained feature-based social in_uence evaluation in online social networks. IEEE Trans. Parallel Distrib. Syst., 25(9): 2286-2296.
[9] Kamvar SD, Schlosser MT, Garcia-Molina H. (2003) The eigentrust algorithm for reputation management in P2P networks. in Proc. WWW, Budapest, Hungary, pp. 640-651
[10] Levien R, Aiken A. (1998) Attack-resistant trust metrics for public key certification. in Proc. USERNIX ATC, San Antono, TX, USA, pp. 229-242.
[11] Liu G, Chen Q, Yang Q, Zhu B, Wang H, Wang W. (2017) Opinionwalk: An ef_cient solution to massive trust assessment in online social networks. in Proc. INFOCOM, Atlanta, GA, USA, pp. 1-9.
[12] Xing L, Wang H, Wang C, Wang Y. (2012) BDD based two party trust sensitivity analysis for social networks. Int. J. Secur. Netw, 7(4): 242-251.
[13] Zaitseva E, Levashenko V, Kostolny J (2015) Application of logical differential calculus and binary decision diagramin importance analysis. Eksploat. Niezawodn., 17(3): 379-388.
[14] Tavakolifard M, Herrmann P, Knapskog SJ. (2009) Inferring Trust Based on Similarity with TILLIT. International Federation for Information Processing, IFIP AICT 300, pp. 133-148.
[15] http://www.trustlet.org/wiki/Advogato_dataset