Electronic Research Archive

http://www.aimspress.com/journal/ERA

ERA, 32(7): 4378–4397.
DOI: 10.3934/era.2024197
Received: 04 March 2024
Revised: 18 June 2024
Accepted: 21 June 2024
Published: 11 July 2024

Research article

Intelligent recommendation algorithm for social networks based on improving a generalized regression neural network

Gongcai Wu*

Hangzhou Vocational & Technical College, Hangzhou 310018, China

Correspondence: Email: 2005010011@hzvtc.edu.cn.

Abstract: In recent years, the vigorous development of the Internet has led to the exponential growth of network information. The recommendation system can analyze the potential preferences of users according to their historical behavior data and provide personalized recommendations for users. In this study, the social network model was used for modeling, and the recommendation model was improved based on variational modal decomposition and the whale optimization algorithm. The generalized regression neural network structure and joint probability density function were used for sequencing and optimization, and then the genetic bat population optimization algorithm was used to solve the proposed algorithm. An intelligent recommendation algorithm for social networks based on improved generalized regression neural networks (RA-GNN) was proposed. In this study, three kinds of social network data sets obtained by real crawlers were used to solve the proposed RA-GNN algorithm in the real social network data environment. The experimental results showed that the RA-GNN algorithm proposed in this paper could implement efficient and accurate recommendations for social network information.

Keywords: generalized regression neural network; social network; smart recommendations; whale optimization algorithm; bat population optimization algorithm; genetic algorithm

1. Introduction

Network security is the cornerstone of computing a power network. Only by guarding against serious security risks such as network attacks and data privacy leaks can computing network services be secure and reliable. With the rise and popularization of various network social platforms, a large amount of personal and social relationship information has been accumulated in social networks, forming a kind of large-scale data with important commercial value. However, the direct release of such data is easy to lead to the disclosure of individual privacy, which will bring troubles to people's lives and even cause major economic losses. Therefore, the issue of intelligent recommendation published for social network data has attracted increasing attention, and its related research has also become the focus of academic attention. Due to the exponential growth of social media information, users can quickly and effectively obtain the required information, so the recommendation system based on social information came into being. Through the analysis of social relations, the introduction of social trust relationship into the recommendation system can greatly improve the accuracy and coverage rate of recommendation. Complex networks are gradually becoming a popular social relationship analysis tool. These studies on improving recommendation effectiveness by analyzing social relationships are based on the premise that users tend to establish social relationships with people with similar preferences [1]. Today's cyberattacks typically consist of multiple carefully planned, imperceptible steps that cause serious damage to a company [2]. In recent years, with the rapid development of communication network and the popularity of mobile devices, network social platforms are favored by the public. People can easily make friends and interact on platforms such as Weibo and Instagram to share interesting experiences with others. These platforms are collectively referred to as social networks [3,4].

Users visit the points of interest they are interested in and post them on the network through mobile devices, thus generating a large number of user sign-in information [5]. These sign-in information contains a lot of valuable content, such as sign-in time, place warp and weft [6,7]. The use of these data to make personalized recommendation system came into being, such a system cannot only save users the time of complicated inquiries, but also help businesses to improve the service of existing points of interest or explore new points of interest for shop location, so as to bring better economic benefits. Due to the huge number of user places, the number of places contained in the user's check-in information is far less than the total number of users, and some user places have not been discovered; thus, the check-in data is very sparse, which will lead to a large error when measuring similarity, affecting the recommendation accuracy rate [8]. Second, the user's check-in information only shows some places that the user has been to, but it does not directly give the user's preference for the visited users. Since the check-in information provides only implicit feedback, from the user's point of view, the recommendation system needs to analyze the user's check-in frequency and influencing factors, use user similarity to predict the user's preferences, and push the interest points that are in line with the preferences [9]. Starting from the point of interest, the system uses the geographic-society-review relationship to construct the user relevance matrix and cluster to obtain the representative recommendation list of the point of interest itself.

At present, news media, Web pages, learning resources, and other fields are applying personalized recommendation technology to improve the personalized level of information service [10]. The existing recommendation systems mostly use information retrieval technology and filtering technology to filter and recommend data sets in their respective fields of application, which seems to solve the problem of information overload; however, the personalized level of knowledge recommendation is not high [11]. Take the electronic literature in virtual knowledge community as an example, most users search the target literature by keyword. Due to the limitation of the input field content and quantity, information unrelated to the target literature is usually displayed or key information is omitted. Even if the advanced search method is used to query the target literature, it is inevitable that irrelevant literature will appear, and users need to obtain the target literature through manual screening [12]. The existing personalized recommendation strategy cannot accurately collect user profile information, and cannot track the change of user preferences and interests in time under the background of continuous expansion and update of information resources. Therefore, the precision of personalized recommendation needs to be improved. More importantly, the existing search system can only present the superficial information resources required by users, but some potential information that users are interested in and hidden behind the literature is not recommended, such as researchers with similar research topics, novel and related research topics, and other deeper knowledge, which seriously limits the depth and breadth of knowledge recommendation. Hindering the process of exchange and communication among users with the same interest and creating new knowledge. There are some research gaps in the field of intelligent recommendation algorithms. For example, existing recommendation algorithms often assume that users' interests are static, but in fact, users' interests change over time; the data of users in intelligent recommendation systems is usually analyzed and shared, and how to better protect users' privacy while implementing algorithms is a problem that needs to be considered at present; recommendation algorithms realize recommendation by analyzing users' historical behavior data, but how to achieve recommendation for unlogin users is also a problem that needs to be studied.

2. Literature review

As a method with intelligent characteristics, the recommendation system can effectively alleviate the current problem of information overload by meeting the needs of potential users and taking the initiative to recommend. In recent years, recommendation system has been widely concerned and applied in many fields, such as e-commerce, information retrieval, intelligent transportation, etc., and People's Daily life more and more closely combined [13]. Compared with the traditional service recommendation, nowadays researchers are faced with more complex and changeable user preferences and scenarios [14]. These data describe the entity objects in the network from multiple angles, such as users, projects, etc., and point out the relationship between entities, such as user social relations, project attributes, content mentioned by users, etc. [15,16]. These data relationships reflect the attributes of network entities, such as user social information, historical behavior information and project attribute information. Accurate modeling of network entities can effectively improve the performance of the recommendation system [17].

The classical recommendation model generally models the relational data as a relational network or bipartite graph [18], represents the network through the adjacency matrix, and analyzes the data through matrix analysis and other methods. However, the available data in the recommendation system is often sparse. In the traditional method, the user's rating information and social information are high-dimensional and sparse user feature vectors. Because there are a lot of empty values in such feature vectors, the common similarity measurement method cannot calculate the relationship between users effectively [19,20]. The sparse relational data makes it impossible for traditional methods to accurately describe users and items, which becomes the bottleneck of constrained recommendation system performance. To solve this problem, network representation learning is proposed and applied to network relationship analysis [21]. Network representation learning embeds data, links, and subgraphs in a network graph into a spatially distributed vector of low dimension. The topology information of the network graph is inherited, and the generated embedding vector can be used by the subsequent learning model in the future learning.

The recommendation model and the recommendation system are not exactly the same, and there are some differences between them. Recommendation model is a mathematical model used to predict the user's score or preference for items, which can predict the items that the user may be interested in by analyzing their historical behavior, and thus provide the recommendation service to the user. The recommendation system is a software system based on the recommendation model, which can recommend items that may be interested in to users according to their historical behavior and preferences. Therefore, recommendation model is an important component of recommendation system, but it is not equivalent to recommendation system. The recommendation model is one of the core technologies of recommendation system, Recommendation system is a more comprehensive system.

The collaborative filtering algorithm uses a similarity matrix and generates corresponding preferences based on neighboring users and projects. The main idea of the method is to find correlations or similarities between users or items. However, the recommendation method of collaborative filtering, as the method of mining users' preferences, needs to use the corresponding historical data, and it has problems such as poor scalability and cold startup [22]. The poor performance of collaborative filtering based algorithms in scalability is mainly due to the limitation of algorithmic ideas. Collaborative filtering algorithms need to find and discover the user's nearest neighbors. However, with the increase in user or project information, this kind of similarity calculation will greatly increase the burden of the system and reduce the efficiency of the recommendation system [23]. The cold start problem is also subject to the basic idea of the algorithm, and because there is no historical data, it cannot find the corresponding neighbors, so the recommendation system will not be able to recommend such users. Although the collaborative filtering recommendation system is limited by its own idea, it gets a lot of development, and the model-based collaborative filtering method has contributed to the development of collaborative filtering to a large extent. The idea is to find a corresponding model to deal with the relationship between users and projects. The type of model can be combined with current technological developments, including methods such as matrix decomposition, machine learning, neural networks, graphs, and other models.

The hybrid recommendation method, as the name suggests, is a method that combines several types of recommendation methods in some way to improve the problem that some methods have when used separately. It was first used to blend collaborative filtering and content-based methods to reduce the technical shortcomings of traditional recommendations and improve precision performance [24]. However, with the continuous development of technology, the hybrid method is no longer limited by the corresponding limitations, and the recommendation quality is improved by extracting the advantages of different models. However, the mixing of models also brings the corresponding problems. The complexity of the recommendation model makes the method have the problems of insufficient flexibility, high complexity, low recommendation precision, and limited results. The main problem of mixed recommendation comes from the complexity of the model [25]. In theory, the recommendation quality should be better than any single recommendation algorithm. Just like the regular term of the loss function, the structure of the model needs to be limited, and it is not an endless stacking of excellent recommendation models that will produce good results. The complexity of the model will increase the cost of recommendation and the consumption of performance. Only by mastering the corresponding balance can the quality of recommendation be effectively improved.

In the recommendation algorithm based on collaborative filtering, the final recommendation effect is often poor because of sparse user scoring matrix and cold startup of the system. In order to alleviate this problem, researchers try to integrate social information into the recommendation system, and propose social recommendation to capture more user characteristic information and improve the recommendation accuracy of the system. Some scholars have introduced the idea of matrix decomposition into social recommendation, and decomposed users' social network matrix and score matrix at the same time, which to some extent solved the problem of sparse data leading to poor recommendation effect [26]. Some scholars put forward the Social MF model, which uses matrix decomposition to introduce the trust propagation mechanism in social networks into the recommendation model [27]. On the basis of SVD++, some scholars proposed a Trust SVD based on social networks, which also considered the social relationship displayed by users and the implicit rating information. Some scholars emphasize the influence of strong and weak relationships in social networks on social recommendation, and use probability matrix decomposition method to distinguish and learn strong and weak social relationships [28]. With the development of graph neural networks, graph structures have shown great potential in modeling social recommendation. Some scholars proposed the Graph Rec model based on the graph convolutional neural network, which simultaneously captures the interaction and opinions in the user-item graph and uses the attention mechanism to aggregate the embedded representation of users and items [29].

3. Model construction

Variable mode decomposition can decompose the fault information and effectively avoid the mode aliasing phenomenon, which is beneficial to analyze the fault information and thus help to realize the feature extraction of the fault signal. Variational mode decomposition realizes the adaptive decomposition of the signal by constructing the variational problem and solving the variational problem, so the eigenmode functions in different frequency bands can be obtained after the signal is decomposed. In a variational model, k represents the number of basis functions or the index of the basis function, and t represents the independent variable, usually time. The variational model shall be constructed as shown in Eq (1). The quadratic penalty factor α and the Lagrange multiplier operator λ are introduced into the above equation to find the optimal solution of the above problem, f represents a coefficient of the Lagrange multiplier that balances the influence between the objective function and the constraints, and the optimal solution after the updated unconstraint condition is shown in Eqs (2) and (3), respectively. The saddle point of the expression obtained in the above process is taken as the optimal solution of the variational model, that is, the variational mode decomposition of the fault signal is realized.

$$IMF(\cdot) = \min_{\{u_1,u_2,\cdots,u_k\},\{\omega_1,\omega_2,\cdots,\omega_k\}} \left(\sum\nolimits_k \left\| \partial_t \left\{ \left[\delta(t) + \frac{j}{\pi t} \right] u_k(t) \right\} e^{-j\omega_k t} \right\|_2^2 \right) \tag{1}$$

$$\hat{u}_k^{n+1}(\omega) = \frac{f \sum_{i \neq k} \hat{u}_i(\omega) + \frac{\hat{\lambda}(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k)^2} \tag{2}$$

$$\hat{\lambda}^{n+1}(\omega) = \hat{\lambda}^n(\omega) + \tau * \hat{f}(\omega) - \tau * \sum\nolimits_k \hat{u}_k^{n+1}(\omega) \tag{3}$$

Because the variational mode decomposition algorithm needs to preset the decomposition number K when decomposition the fault signal, and the size of K is different, the final decomposition effect is also quite different, and if K is too small, the mode aliasing phenomenon may occur. If K is too large, there may be an over decomposition phenomenon, and the size of the quadratic penalty factor α has a large effect on the bandwidth of the mode component of the variational mode decomposition. Therefore, in this study, the whale algorithm is used to optimize the decomposition number K and the quadratic penalty factor α to find the optimal K and α in the calculation examples, so as to improve the effect of variational mode decomposition. When using whale algorithm to optimize parameters, it is necessary to determine the fitness function to evaluate whether the optimization result is optimal. Sample entropy is used to evaluate whether the optimization result is optimal. The smaller the sample entropy is, the smaller the probability that the decomposed sequence will generate a new pattern, that is, the smaller the frequency component in the sequence, the smaller the possibility of mode aliasing, and the decomposition effect will be relatively better; therefore, sample entropy of each component after decomposition is calculated, and the function of solving the minimum entropy of all sample entropy is taken as the fitness function. Since the humpback whale recognizes the position of the prey and rotates around it, but the position of the optimal solution in the optimized search space is not certain, the best candidate solution is the target prey, and the best search agent is repeatedly updated and defined, as shown in Eqs (4) and (5).

$$\vec{Y} = \left|\vec{D} \odot \vec{X}^*_{t+1} - \vec{X}_t\right| \tag{4}$$

$$\vec{X}_{t+1} = X^*_t - \vec{C} \odot \vec{Y} \tag{5}$$

Within it, $|\cdot|$ represents the absolute value of the vector element, $\odot$ represents the product of elements of two vectors (that's Hadamard matrix). $\vec{C}$ and $\vec{D}$ are coefficient vectors, and $\vec{X}^*$ is the position vector of the best solution obtained in the corresponding iterations t and t + 1. Vectors $\vec{C}$ and $\vec{D}$ are calculated as described in Eq (6).

$$\vec{C} = 2\vec{a} \odot \vec{r} - \vec{a}, \vec{D} = 2\vec{r} \tag{6}$$

Within it, $\vec{a}$ Linearly reduced from 2 to 0 during the iteration, r is the random vector in the $[0,1]$. The whale optimization algorithm has two design methods for the mathematical model of the humpback whale Bubble-net attack method. Moreover, by assigning random values to C in the interval $[-1,1]$, the new position of the search agent is available anywhere within its original position and within the current best agent position. The spiral position is updated by calculating the distance between the whale and the prey and providing a spiral equation between it, as shown in Eq (7).

$$\vec{X}_{t+1} = e^{bl} \cos(2\pi l) * \vec{Y'} + \vec{X}^*_t \tag{7}$$

Within it, $\vec{Y'} = \left|\vec{X}^*_t - \vec{X}_t\right|$ represents the distance of the whale from its prey (that is the best solution obtained currently), the constant b defines the shape of the logarithmic spiral, and I is the random number in the interval $[-1,1]$. Because the humpback whale moves along a spiral path in a reduced circle, the contraction encirclement mechanism and the spiral model in the whale position update can be assumed to be Pe% and 1-Pe%, respectively. Therefore, the mathematical model is shown in Eq (8).

$$\vec{X}_{t+1} = \begin{cases} \overrightarrow{X_t}^* - \vec{C} \odot \vec{Y} & if\ p < Pr \\ e^{bl} * cos(2\pi l) * \vec{Y}' + \vec{X}_t^* & if\ p \geq Pe \end{cases} \tag{8}$$

Within it, p is the random number drawn from the interval $[0,1]$. Humpback whales were searched randomly according to each other's location. Therefore, a random value of C greater than 1 or less than -1 is used to push the search agent away from the reference. Contrary to the attack phase, the location of the search agent is formed according to the randomly selected search agent, and is not updated by the current best search agent. This process and $|\vec{\mathrm{C}}| > 1$ help the whale optimization algorithm to run the global search, as shown in Eq (9). Where, $\vec{\mathrm{X}}_{\mathrm{rand}}$ is the random position vector chosen from the current population, the whale optimization algorithm first uses a series of random solutions and updates its location in the search agent according to a randomly selected search agent or the current best solution. In the iterative process of updating the search agent location, a random search agent is selected if $|\vec{\mathrm{C}}| > 1$, while the best solution is chosen if $|\vec{\mathrm{C}}| < 1$. Given the p value, the whale optimization algorithm can alternate between spiral motion or circular motion, and the variational mode decomposition algorithm improved by the whale optimization method can be represented as shown in Algorithm 1.

$$\{\vec{Y}, \vec{X}_{t+1}\} = \{|\vec{D} * \vec{X}_r - \vec{X}|, |\vec{X}_r - \vec{C} * \vec{Y}|\} \tag{9}$$

The time complexity of this algorithm mostly depends on the candidate set size, number of iterations, number of decomposition K, and the sample entropy calculation time. Therefore, the time complexity can be expressed as O (C * N * K * S * T), where C is the size of the candidate set, N is the number of iterations, K is the number of decomposition, S is the calculation time of sample entropy, and T is the fitness calculation time of a search node. In terms of spatial complexity, the major operations of this algorithm are multiplication, addition, and comparison of matrices or vectors. The spatial complexity of these operations is O (n) or O (m), where n is the signal length and m is the number of parameters in the search space. Therefore, the spatial complexity of this algorithm is O (nm).

Algorithm 1: Improved variational mode decomposition algorithm for improved whale optimization

```
Initialize the node population X_i(i = 1,2, ..., n)
Set the parameters such as the maximum number of iterations
The combination of decomposition number K and penalty factor α is taken as node positions, and the parameter combination (K, α) is randomly generated
Compute the current fitness for each search node
Make X_1^* the optimal search node
Conduct the variational mode decomposition of the fault signal according to different parameter combinations and calculate the sample entropy value for each component
for t = 1 to T then
    for each search node do
        update parameter a, C, D, l, p
    if p < Pe, |C| < 1 then
        Update the location of the current node
            else if |C| ≥ 1
        Select a random search agent X_rand to update the location of the current fitness
    end if
    else if p ≥ Pe then
    Update the location of the current search and calculates the new fitness for each search agent
    end if
if the new fitness values are smaller than the previous fitness values then X_t^* = X_t
end for
return X_t^*
```

I use the social network model as the basis for building the recommendation model. By combining the variational modal decomposition and whale optimization algorithm, the recommendation model is optimized and improved. The generalized regression neural network structure and the joint probability density function are used to sort and optimize the recommended results. Finally, the genetic bat population optimization algorithm is used to solve and optimize the proposed social network intelligent recommendation algorithm based on the improved generalized regression neural network.

In addition, I use the generalized regression neural network structure and the joint density function to sort and optimize the algorithm, which mainly involves the following two aspects: First, the generalized regression neural network can be used to deal with the complex data relationship in the ranking algorithm due to its powerful nonlinear mapping ability and learning speed. The structure of generalized regression neural network allows it to efficiently extract key information from the input data and obtain the output results through computation in the network layer. Second, the joint density function plays a key role in the algorithm sorting. The joint density function is able to describe the joint probability distribution of multiple random variables, and thus can reveal the dependence and correlation among the variables. In the sorting problem, these dependencies and correlations may affect the outcome of the ordering. Using the joint density function, we can more accurately understand the intrinsic links between the data and optimize the sorting algorithm accordingly. In conclusion, the algorithm sorting and optimization using generalized regression neural network structure and joint density function can make full use of the advantages of both and improve the performance and accuracy of the sorting algorithm.

4. Algorithm design

The generalized regression neural network structure is based on the radial base neural network, with strong nonlinear mapping ability and learning speed. It contains four layers, in which the input variables are directly transmitted to the mode layer by the input layer, and the transfer function can be expressed as shown in Eq (10).

$$p_i = exp\left[-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2}\right], \quad i = 1,2,\cdots,n \tag{10}$$

Within it, X represents the input learning sample; X_i is the training sample; p_i is the exponential molecule is the sum of Euclidean distance squares between the learning sample and the training sample; and σ is the Gaussian function standard deviation. The output layer output sample is $Y = [y_1, y_2, \cdots, y_n]$, the output layer neuron output can be represented as shown in Eq (11).

$$y_j = \frac{\sum_{i=1}^{n} P_i}{\sum_{i=1}^{n} y_{ij} P_i}, \quad j = 1,2,\cdots,k \tag{11}$$

Within it, P_i is the output of neurons in each mode layer; S_{Nj}, S_D is the sum of two types of neurons; and y_{ij} is the jth element of Y_i. Due to its flexible network structure, high fault tolerance and robustness, the generalized regression neural network can get better prediction and recommendation effect in the social network environment for the problems of few samples and poor accuracy of some sample data [30]. A reasonable volume attenuation coefficient and search frequency can effectively balance the optimization accuracy and convergence speed of the algorithm, and it is necessary to adjust the volume attenuation coefficient and search frequency constantly in the algorithm simulation. Compared with other particle swarm algorithms, the simple model, fewer parameters, and fast convergence have become the prominent advantage of the bat algorithm. In the process of algorithm optimization, the attraction of super bats to other bats in the population will decrease significantly due to the lack of variation mechanism. The cross-variation operator was used to enrich the diversity of bat populations.

The neurons in the input layer correspond to multiple influencing factors, and the number of neurons in the pattern layer corresponds to multiple groups of training sample species [31]. In terms of increasing the diversity of bat populations, the most common and basic single-point cross operator and fundamental bit variation operators were used to optimize the bat algorithm, respectively. The single-point crossover operator refers to randomly setting the crossover points in the parent chromosome coding string of pairwise pairs of individuals, and then exchanging the two-segment chromosomes of the two parents at the crossover point. Each crossover operation of the matched parent chromosomes produces the same number of offspring chromosomes, and depending on the selection of the intersection, the two offspring chromosomes from the two parent chromosomes have different lengths of loci. The basic bit mutation operator is to randomly generate a random number r' within [0,1] for each bit on the individual coding locus to determine whether the bit is calculated according to the size of the mutation probability p_m and r', and the gene value on the locus of $p_m > r'$ is used as the mutation operation. To improve the topological structure features of the generalized regression neural network,

Let the $x = [x_1, x_2, \cdots, x_j]^T$ in the multidimensional vector j be the input vector of the procedure, the corresponding output vector is y, the joint probability density function of the random variables x and y is $f(x, y)$. Since the theoretical basis of GRNN is nonlinear regression analysis, by giving the value of x, the conditional mathematical expectation of y is calculated for regression. GRNN estimates the sum of the joint probability density function, to build an estimated probability model. By training the input-output set, the PDF estimator is constructed using the non-parametric density estimation method. For a given input vector x, assuming that the estimated function is continuous and smooth, the expected value of the estimated y is as shown in Eq (12).

$$E[y|x] = \frac{\int_{-\infty}^{\infty} v(2\pi)^{\frac{p+1}{2}} \sigma^{(p+1)} * k * \sum_{i=1}^{k} exp\left[-\frac{(x-x_i)^T(x-x_i)}{2\sigma^2} * \left[-\frac{(y-y_i)^2}{2\sigma^2}\right]\right] dv}{\int_{-\infty}^{+\infty} f(x,y)dy} \tag{12}$$

Within it, $f(x, y)$ is a function of the continuous probability density function. x_i , y_i is the i th sample value of the random variables x and y, σ is the smoothing parameter, p is the dimension of the random variable x, and k is the number of samples. The topology of generalized regression neural network has network structure such as input layer, pattern layer, sum layer, and output layer. In this study, the following improvements are made. First, the sample is input to the input layer, the number of nodes in the input layer is equal to the dimension p of the input vector, and then the elements of the input vector are transferred to the pattern layer. The mode layer transfer function is shown in Eq (13).

$$t_i = exp\left(-\frac{D_i^2}{2\sigma^2}\right), \quad i = 1,2, \cdots, k \tag{13}$$

However, the summing layer has two types of nodes. If only a single neuron is included, the output of all pattern layer neurons is arithmetically summed, the connection weight of each neuron between the neuron in the pattern layer and the neuron is 1, and the transfer function is $s_D = \sum_{i=1}^{n} P_i$. If the remaining nodes are included, the output of all pattern layer neurons is weighted and summed, the transfer function of the summing neuron is $s_j = \sum_{i=1}^{n} y_{ij} P_i$, $j = 1,2, \cdots, 1$. Where, y_{ij} is the connection weight between the ith neuron in the pattern layer, and the jth summation neuron in the summation layer is the jth element in the ith output sample. The number of nodes in the output layer is equal to the dimension m of the output vector in the learning sample. In the study, genetic algorithm (GA) and bat algorithm are combined to solve problems, as shown in Algorithm 2 (that is, this study proposes social network intelligent recommendation algorithm based on improved generalized regression neural network (RA-GNN)).

Algorithm 2: Genetic-bat optimized selection algorithm

Begin
Learning sample and prediction sample data normalization
    Generate the initial candidate population (create a set of binary vectors)
  Initialize the algorithm parameters A_0, r_0, α, γ and the maximum number of iterations $N_{\max}$;
    Initialize the cross probability p_c and the variant probability p_m
      While (Not the maximum iteration) **or** (No change in fitness for successive iterations) **do**
        Evaluate each candidate (train the classifier and calculate the loss function using the selected feature)
        Random selection of the solutions for the next generation
        Apply crossover and variation
      End while
    Calculated the initialize population fitness, and recorded the initial optimal solution fitness function values $f_{\min}$, position, and speed
        Individual bat values under position I were assigned to B_{est}
      The parts of the parental chromosome are exchanged at the paired individual intersection according to the crossover probability p_c
      Generate uniform distributed random numbers R_{rand}, and calculate the fitness of all bat individuals and rank them
Return the optimal solution (binary vector) as the smoothing factor of the generalized regression audit network
End
return B_{est}

The complexity of this algorithm depends on the candidate set size, number of iterations, number of features, and individual chromosome length. Therefore, the time complexity is O (C * N * F * L * T), the space complexity is O (C * N * F * L), where C is the size of the candidate set, N is the number of iterations, F is the number of features, L is the length of an individual chromosome, and T is the training time of a classifier and the time to calculate the loss function.

5. Simulation examples

5.1. Experimental design and data description

The proposed algorithms are evaluated here, and the practices are in [32–34]. Three real social network datasets were used in the experiment (as shown in Table 1). I use Python software to take 30 recent hot comment user nodes as the initial nodes, and take the above variable as samples of media image crisis of individuals and organizations with strong representation, and the Chinese and English user data sets of Sina Weibo and Facebook were respectively crawled as the basic data of the experimental simulation (the crawling time was from May 13, 2022 to August 22, 2022). I treat each user as a node, using edges between nodes to represent the relationships between users. In this study, 30 influential users and their friends list were selected as the initial nodes of social networks, so as to generate simple social networks. The research team implemented the proposed algorithm and the relevant algorithms for comparison on Tensorflow1.5.1. The experiment was conducted in groups, and the data were divided into 10 groups by cross-validation, that is, the data set was divided into 10 equal parts, one set of data was selected as the test set at a time, and the other groups as the training set, and the final average value was obtained. All data was saved in CSV format in the MySQL database for data processing. For each social network dataset, 10% of each user rating data was randomly selected as the test set using the Rapidminer data mining tool, and the remaining 90% user data was used as the training set. The movie Review network built the knowledge map of Wikipedia and Baidu Encyclopedia for the experiment. Python was used as the development language, and all experiments were completed in the environment of Python3.7.2 and tensorflow1.13.0. It runs on two Linux-based servers (Intel Xeon processors (34 GHZ) with 64 GB of RAM), each with an 8-core CPU, two NVIDIA Titan X Gpus, and 100 GB of RAM. Since the results of the public opinion control model in the experiment may be different during each run, the evaluation results are set as the average value after 500 iterations, and the standard deviation of the operation is 1.527.

Table 1. Social network dataset.

Network serial number	Social network name	Type	Number of nodes	Number of node boundaries	Average degree	Node Average Path	Cluster coefficient
1	Facebook network	Directed	24,252	319,204	34.350	2.245	0.154
2	Sina Weibo network	Directed	23,910	238,591	41.245	2.519	0.148
3	Movie Reviewnet work	Directed	27,810	582,910	31.501	2.490	0.130

In this study, the proposed social network intelligent recommendation algorithm based on improved generalized regression neural network (RA-GNN) is compared with the standard algorithms such as Degree Centrality (DC), Intermediate Centrality (IC), Closeness Centrality (CNC), Ant Colony Optimization (ACO), Swarm Optimization (SWO), K-Shell Centrality (KSC), and Weighted K-Shell Degree Neighborhood (WKS-DN). Recommendation problems in social networks are often viewed as binary classification tasks, while in the evaluation confusion matrix with two categories, True Positive (TP) represents the number of correctly predicted links, True Negative (TN) represents the number of correctly unpredicted links, False Positive (FP) represents the number of incorrectly predicted links, and the False Negative (FN) represents the number of incorrectly unpredicted links. Based on this, the evaluation indicators of accuracy, accuracy, recall and F-measure can be expressed as shown in Eqs (14)–(17). In addition, two accuracy functions are used [35,36]: Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The MAE is the average of the absolute value of the deviation of all individual observations from the arithmetic mean. The RMSE is the square root of the average of the square of the deviation of all individual observations from the arithmetic mean. The specific calculation methods are shown in Eqs (18) and (19), respectively.

$$Precision = \frac{TP}{TP+FP} \tag{14}$$

$$Accuracy = \frac{TP+TN}{P+N} \tag{15}$$

$$Recall = \frac{TP}{TP+FN} \tag{16}$$

$$F - measure = \frac{2*precision*recall}{precision+recall} \tag{17}$$

$$\mathrm{MAE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}|\mathrm{f_i} - \mathrm{y_i}| \tag{18}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(observed_t - predicted_i)^2} \tag{19}$$

In this paper, the graph coloring technique is used to propose a social network intelligent recommendation algorithm based on improved generalized regression neural network (RA-GNN). Coloring a graph G is defined as painting each vertex a color, and any adjacent vertices are painted differently: if the vertices of G can be colored with kcolors, that is, G is colorable at k points; if G is colorable at point k, but not at point k-1, then G is said to be a k-color graph, and k is the color number of G, denoted by x(G), which is the minimum value of k that makes G colorable. Based on this, this study proposes a graph coloring solution technique for coloring assignment among the vertices of network graphs, which is essentially a linear decomposition of the graph to color16. To make the proposed solution algorithm suitable for large-scale graphs, a hybrid technique based on decomposition and heuristic methods is proposed, specifically as shown in Algorithm 3.

Algorithm 3: Social network graph color solution algorithm

Import: grap $G = (V, E), S, A, P, \tau, k$
Output: upper and lower bound $B(+)/B(-)$
1: Set up $B(\cdot) \leftarrow \emptyset$
2: Use the graph coloring algorithm to calculate upper and lower bounds $B(+)/B(-)$
2: **while** $G(\cdot) = \sum x_{ik} * x_{jk} = 0, x_{ik}, x_{jk} \in \{0,1\}$ **do**
3: **if** no solution makes the lower bound $B(-) = B(-) + 1$
4: The upper bound $B(+)$ is calculated
5: **end if** $B(-)! = B(+)$
6: **end**
7: return $B(+)/B(-)$

The time complexity of this algorithm depends on the graph coloring algorithm used, so the time complexity is O (n^2), where n is the number of vertices. The space complexity is O (1), because it uses only the additional space at the constant level to store the intermediate results, without increasing with the size of the input.

5.2. Experimental results

While Table 2 reports the results of the area value under the curve of the proposed recommendation algorithm based on improved generalized regression neural network (RA-GNN) and other benchmark methods in the social network dataset. I found that the proposed recommendation algorithm based on RA-GNN has better experimental results in the real social network datasets. Table 3 reports the results of the average accuracy values of the proposed recommendation algorithm based on RA-GNN and other benchmark algorithms in the social network datasets. The results show that the proposed recommendation algorithm based on RA-GNN has high average accuracy values in all social network datasets.

Combined with literature, two precision functions are used: Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Table 4 reports the results of the MAE and RMSE values of the proposed RA-GNN improved generalized regression neural network algorithm and other benchmark algorithms in different social network topologies. The higher the MAE and RMSE values, the lower the accuracy of the prediction optimization algorithm. According to Table 5, the proposed recommendation algorithm based on RA-GNN generally outperforms the other methods, this is due to the ability of RA improved generalized regression neural network algorithms to respond quickly and adjust in real time to optimize social networks.

Table 2. Area under the curve values in the different methods for each dataset.

Cross validation level	Dataset name	Social network optimization algorithm			
		DC	IC	CNC	ACO
2-fold	Facebook network	0.1452	0.2190	0.2491	0.3294
	Sina Weibo network	0.1592	0.2301	0.2019	0.3001
	Movie Review network	0.1230	0.2481	0.2385	0.3850
4-fold	Facebook network	0.1238	0.2984	0.2572	0.3395
	Sina Weibo network	0.1394	0.2385	0.2398	0.3182
	Movie Review network	0.1205	0.2039	0.2185	0.3059
10-fold	Facebook network	0.1504	0.2428	0.2385	0.3295
	Sina Weibo network	0.1648	0.2385	0.2494	0.3128
	Movie Review network	0.1248	0.2419	0.2019	0.3204
Cross validation level	Dataset name	Social network optimization algorithm			
		SWO	KSC	WKS-DN	RA-GNN
2-fold	Facebook network	0.4384	0.4383	0.5242	0.8733
	Sina Weibo network	0.4292	0.4891	0.4790	0.8291
	Movie Review network	0.3294	0.4292	0.4951	0.8402
4-fold	Facebook network	0.4355	0.5103	0.5255	0.8390
	Sina Weibo network	0.4202	0.4380	0.5096	0.7929
	Movie Review network	0.4239	0.4289	0.5200	0.8109
10-fold	Facebook network	0.4292	0.4780	0.5191	0.7985
	Sina Weibo network	0.4110	0.4381	0.5039	0.8792
	Movie Review network	0.4039	0.4102	0.5393	0.8330

Note: The values shown in bold all indicate the good performance of the corresponding algorithm.

Table 3. Average accuracy values of each dataset across the different methods.

Cross validation level	Dataset name	Social network optimization algorithm			
		DC	IC	CNC	ACO
2-fold	Facebook network	0.1295	0.1842	0.3382	0.3182
	Sina Weibo network	0.1315	0.1728	0.3194	0.3294
	Movie Review network	0.1248	0.1984	0.3102	0.3294
4-fold	Facebook network	0.1342	0.1829	0.3201	0.3209
	Sina Weibo network	0.1325	0.1920	0.3284	0.3396
	Movie Review network	0.1240	0.2043	0.3095	0.3102
10-fold	Facebook network	0.1492	0.1928	0.3211	0.3859
	Sina Weibo network	0.1311	0.1829	0.3329	0.3294
	Movie Review network	0.1429	0.1782	0.3102	0.3497
Cross validation level	Dataset name	Social network optimization algorithm			
		SWO	KSC	WKS-DN	RA-GNN
2-fold	Facebook network	0.4891	0.5199	0.5343	0.7129
	Sina Weibo network	0.4392	0.5920	0.5190	0.7491
	Movie Review network	0.4209	0.5829	0.5291	0.8352
4-fold	Facebook network	0.4829	0.5529	0.5404	0.7580
	Sina Weibo network	0.4282	0.5389	0.5192	0.7692
	Movie Review network	0.4729	0.5270	0.5380	0.8738
10-fold	Facebook network	0.4829	0.5327	0.5389	0.8693
	Sina Weibo network	0.4281	0.5192	0.5192	0.8231
	Movie Review network	0.4389	0.5637	0.5029	0.8402

Note: The values shown in bold all indicate the good performance of the corresponding algorithm.

Table 4. Comparative results of the social network optimization algorithm.

	Indicator	DC	IC	CNC	ACO
Actual value	MAE	0.8492	0.8291	0.7109	0.6483
	RMSE	0.8519	0.8692	0.7281	0.6891
Optimal value	MAE	0.8293	0.8029	0.6930	0.6291
	RMSE	0.8391	0.8192	0.7195	0.6310
	Indicator	SWO	KSC	WKS-DN	RA-GNN
Actual value	MAE	0.5302	0.5020	0.5210	0.3120
	RMSE	0.5394	0.5492	0.5391	0.3422
Optimal value	MAE	0.5029	0.4980	0.5032	0.2910
	RMSE	0.5140	0.5029	0.5105	0.3019

Note: The bold place indicates that the algorithm is relatively optimal under this parameter condition.

Table 5. Comparison results of algorithms in different social network datasets.

Dataset name	DC	IC	CNC	ACO
Facebook network	2422.321	819.849	492.193	849.232
Sina Weibo network	3244.831	1079.124	979.124	1109.753
Movie Review network	7491.293	793.351	692.429	1032.23
Dataset name	SWO	KSC	WKS-DN	RA-GNN
Facebook network	1423.244	242.402	435.317	120.439
Sina Weibo network	1592.890	310.985	782.295	142.795
Movie Review network	1230.402	248.312	591.940	125.951

Note: The values shown in bold all indicate that the corresponding algorithm performs well.

6. Conclusions

In this study, the social network model was used for mode, and the recommendation model was improved based on variational mode decomposition and whale optimization algorithm, and the generalized regression neural network structure and joint probability density function were used for ranking and optimization; then, the genetic bat population optimization algorithm was used to solve the proposed algorithm, a social network Intelligent recommendation algorithm based on RA-GNN. This study also used three social network datasets obtained from real reptiles, to solve the proposed RA-GNN algorithm in a real social network data environment. The experimental results show that the proposed RA-GNN algorithm can make an efficient and accurate recommendation for social network information.

The recommendation algorithm proposed in this paper (RA-GNN) can effectively make use of user behavior data and social relationship information in social networks, greatly analyze users' evaluation of products, capture users' interests and preferences, and achieve good recommendation performance. The algorithm can be applied to the recommendation of music, movies, videos and e-commerce. Compared with the existing recommendation algorithms, the proposed recommendation algorithm can recommend more interesting content to users more accurately and efficiently. RA-GNN can not only greatly improve user satisfaction, save user screening time, and increase user stickiness, but also bring good additional benefits to Internet merchants. Through the application of personalized recommendation algorithms, the video click rate can be increased by about 50%, which can improve merchants' income by about 30% and contribute to the improvement of the social economy. In the academic field, this paper innovates the research of recommendation algorithms and proposes a social network intelligent recommendation algorithm based on improving a generalized regression neural network; this effectively alleviates the problems of poor scalability and cold start of existing recommendation algorithms and can reduce the consumption of recommendation costs and performance. This research promotes the research process in the field of recommendation algorithms and provides new thinking and reference value for the research of recommendation algorithms.

I present the above important findings, though this study has limitations. First, the relationship between users in social networks is usually more complex and uncertain. The ability of the algorithm proposed in this paper to resist uncertainty needs to be improved in the follow-up work. For this problem, the uncertainty of nodes and edges can be considered at the same time in the social network model to further improve the ability of the algorithm proposed in this paper to resist uncertainty. Second, the scope of the data used in this paper in evaluating the algorithm is not comprehensive enough and has limitations. We can explore integrating time, place, organizer, and other information into the framework of the social network recommendation algorithm so as to further improve the performance of the recommendation algorithm and hope to better deal with cold start problems and security threats; and further improve the efficiency and accuracy of the algorithm proposed in this paper. Finally, how to recommend the algorithm proposed in this paper more accurately and efficiently is also the scope of my research. I use particle swarm optimization algorithm and other cutting-edge algorithms to implement recommendation, in order to further improve the accuracy and efficiency of social network information recommendation.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. S. Ajmal, M. Awais, K. S. Khurshid, M. Shoaib, A. Abdelrahman, Data mining-based recommendation system using social networks-an analytical study, *PeerJ Comput. Sci.*, **9** (2023), e1202. https://doi.org/10.7717/peerj-cs.1202
2. A. Jokić, S. Baraković, J. B. Husić, J. Pleho, Partial rule security information and event management concept in detecting cyber incidents, *Int. J. Secur. Netw.*, **16** (2021), 117–128. https://doi.org/10.1504/IJSN.2021.116777
3. Y. Yu, W. Qian, L. Zhang, R. Gao, A graph-neural-network-based social network recommendation algorithm using high-order neighbor information, *Sensors*, **22** (2022), 7122. https://doi.org/10.3390/s22197122
4. B. Lyes, A. Mourad, A. Djamil, A. Sofiane, P2PCF: A collaborative filtering based recommender system for peer to peer social networks, *J. High Speed Networks*, **27** (2021), 13–31. https://doi.org/10.3233/JHS-210649
5. F. Li, K. Wei, Study on social network recommendation service method based on mobile cloud computing, *Int. J. Auton. Adapt. Commun. Syst.*, **14** (2021), 3983–408. https://doi.org/10.1504/IJAACS.2021.119124
6. E. Rajalakshmi, V. Subramaniyaswamy, V. Vijayakumar, L. Ravi, Location-based social network recommendations with computational intelligence-based similarity computation and user check-in behavior, *Concurrency Comput. Pract. Exper.*, **33** (2021), e6106. https://doi.org/10.1002/cpe.6106

7. S. P. Perumal, G. Sannasi, K. Arputharaj, FIRMACA-Fuzzy intelligent recommendation model using ant clustering algorithm for social networking, *SN Appl. Sci.*, **2** (2020), 1704. https://doi.org/10.1007/s42452-020-03486-4
8. R. Chen, Y. Chang, Q. Hua, Q. Gao, X. Ji, B. Wang, An enhanced social matrix factorization model for recommendation based on social networks using social interaction factors, *Multimedia Tools Appl.*, **79** (2020), 14147–14177. https://doi.org/10.1007/s11042-020-08620-3
9. A. Ghorbel, M. Ghorbel, M. Jmaiel, A model-based approach for multi-level privacy policies derivation for cloud services, *Int. J. Secur. Netw.*, **16** (2021), 12–27. https://doi.org/10.1504/IJSN.2021.112836
10. M. E. A. A. Tharwat, D. W. Jacob, M. F. M. Fudzee, S. Kasim, A. A. Ramli, M. Lubis, The role of trust to enhance the recommendation system based on social network, *Int. J. Adv. Sci. Eng. Inf. Technol.*, **10** (2020), 1387–1395. https://doi.org/10.18517/ijaseit.10.4.10883
11. A. Khaled, S. Ouchani, C. Chohra, Recommendations-based on semantic analysis of social networks in learning environments, *Comput. Hum. Behav.*, **101** (2019), 435–449. https://doi.org/10.1016/j.chb.2018.08.051
12. M. Aivazoglou, A. O. Roussos, D. Margaris, C. Vassilakis, S. Ioannidis, J. Polakis, et al., A fine-grained social network recommender system, *Social Network Anal. Min.*, **10** (2019), 8. https://doi.org/10.1007/s13278-019-0621-7
13. S. Bedda, O. Kraa, M. Mohammedi, D. E. Zabia, I. Tegani, M. K. Benbraika, Optimization of passivity-based controller for a hybrid vehicle power source using the gray wolf algorithm, in *2024 8th International Conference on Image and Signal Processing and their Applications (ISPA)*, Biskra, Algeria, IEEE, (2024), 1–7. https://doi.org/10.1109/ISPA59904.2024.10536755
14. Z. Zhang, R. Sun, K. R. Choo, K. Fan, W. Wu, M. Zhang, et al., A novel social situation analytics-based recommendation algorithm for multimedia social networks, *IEEE Access*, **7** (2019), 117749–117760. https://doi.org/10.1109/ACCESS.2019.2934898
15. A. Zare, M. R. Motadel, A. Jalali, Presenting a hybrid model in social networks recommendation system architecture development, *AI Soc.*, **35** (2019), 469–483. https://doi.org/10.1007/s00146-019-00893-z
16. Z. Liu, H. Zhong, Study on tag, trust and probability matrix factorization based social network recommendation), *KSII Trans. Internet Inf. Syst.*, **12** (2018), 2082–2102. https://doi.org/10.3837/tiis.2018.05.010
17. Z. Ding, X. Li, C. Jiang, M. Zhou, Objectives and state-of-the-art of location-based social network recommender systems, *ACM Comput. Surv.*, **51** (2018), 1–28. https://doi.org/10.1145/3154526
18. R. Jia, R. Li, M. Gao, Study on data sparsity in social network-based recommender system, *Int. J. Comput. Sci. Eng.*, **20** (2019), 15–20. https://doi.org/10.1504/IJCSE.2019.103245
19. X. Zhao, Z. Ma, Z. Zhang, A novel recommendation system in location-based social networks using distributed ELM, *Memet. Comput.*, **10** (2018), 321–331. https://doi.org/10.1007/s12293-017-0227-4
20. S. Uribe, C. Ramirez, J. Finke, Recommender systems based on matrix factorization and the properties of inferred social networks, *Discrete Math., Algorithms Appl.*, **16** (2024), 2350052. https://doi.org/10.1142/S1793830923500520

21. C. Xu, A novel recommendation method based on social network using matrix factorization technique, *Inf. Process. Manage.*, **54** (2018), 463–474. https://doi.org/10.1016/j.ipm.2018.02.005
22. D. Margaris, C. Vassilakis, P. Georgiadis, Recommendation information diffusion in social networks considering user influence and semantics, *Social Network Anal. Min.*, **6** (2016), 108. https://doi.org/10.1007/s13278-016-0416-z
23. P. Kefalas, P. Symeonidis, Y. Manolopoulos, Recommendations based on a heterogeneous spatio-temporal social network, *World Wide Web*, **21** (2018), 345–371. https://doi.org/10.1007/s11280-017-0454-0
24. M. Eirinaki, J. Gao, I. Varlamis, K. Tserpes, Recommender systems for large-scale social networks: A review of challenges and solutions, *Future Gener. Comput. Syst.*, **78** (2018), 413–418. https://doi.org/10.1016/j.future.2017.09.015
25. M. G. Campana, F. Delmastro, Recommender systems for online and mobile social networks: A survey, *Online Social Networks Media*, **3–4** (2017), 75–97. https://doi.org/10.1016/j.osnem.2017.10.005
26. Y. Wang, M. Ding, Z. Chen, L. Luo, Caching placement with recommendation systems for cache-enabled mobile social networks, *IEEE Commun. Lett.*, **21** (2017), 2266–2269. https://doi.org/10.1109/LCOMM.2017.2705695
27. G. Rojas, I. Garrido, Toward a rapid development of social network-based recommender systems, *IEEE Lat. Am. Trans.*, **15** (2017), 753–759. https://doi.org/10.1109/TLA.2017.7896404
28. E. A. Mohammed, N. Linge, Cloud based electronic program guide system with integrated social network recommendations, *I. J. e-Educ., e-Bus., e-Manage. e-Learn.*, **7** (2017), 100–110. https://doi.org/10.17706/ijeeee.2017.7.2.100-110
29. T. Li, M. Zhao, A. Liu, C. Huang, On selecting vehicles as recommenders for vehicular social networks, *IEEE Access*, **5** (2017), 5539–5555. https://doi.org/10.1109/ACCESS.2017.2678512
30. L. Luo, J. Dong, W. Kong, Y. Lu, Q. Zhang, Short-term probabilistic load forecasting using quantile regression neural network with accumulated hidden layer connection structure, *IEEE Trans. Ind. Inf.*, **20** (2024), 5818–5828. https://doi.org/10.1109/TII.2023.3341242
31. Y. Zhang, X. Li, Y. Zhang, A novel integrated optimization model for carbon emission prediction: A case study on the group of 20, *J. Environ. Manage.*, **344** (2023), 118422. https://doi.org/10.1016/j.jenvman.2023.118422
32. L. S. Kullappa, R. A. Kumar, R. Kullappa, A study on recommendation systems in location based social networking, *J. Inf. Optim. Sci.*, **41** (2017), 213–229. https://doi.org/10.31341/jios.41.2.6
33. C. Zhao, S. Sun, L. Han, Q. Peng, Hybrid matrix factorization for recommender systems in social networks, *Neural Network World*, **26** (2016) 559–569. https://doi.org/10.14311/NNW.2016.26.032
34. D. Margaris, C. Vassilakis, Exploiting rating abstention intervals for addressing concept drift in social network recommender systems, *Informatics*, **5** (2018), 21. https://doi.org/10.3390/informatics5020021
35. M. A. Zharova, V. I. Tsurkov, Neural network approaches for recommender systems, *J. Comput. Syst. Sci. Int.*, **62** (2023), 1048–1062. https://doi.org/10.1134/S1064230723060126

36. H. Li, X. Ma, J. Shi, Incorporating trust relation with PMF to enhance social network recommendation performance, *Int. J. Pattern Recognit Artif Intell.*, **30** (2016), 1659016. https://doi.org/10.1142/S0218001416590163

AIMS Press