RESEARCH ARTICLE

Network Intrusion Detection Based on a General Regression Neural Network Optimized by an Improved Artificial Immune Algorithm

Jianfa Wu[1], Dahao Peng[2], Zhuping Li[2], Li Zhao[3], Huanzhang Ling[4]*

1 College of Automation, Harbin Engineering University, Harbin, Heilongjiang, China, **2** College of Computer Science and Technology, Harbin Engineering University, Harbin, Heilongjiang, China, **3** College of National Secrecy, Harbin Engineering University, Harbin, Heilongjiang, China, **4** College of Science, Harbin Engineering University, Harbin, Heilongjiang, China

* linghuanzhang@hrbeu.edu.cn

Citation: Wu J, Peng D, Li Z, Zhao L, Ling H (2015) Network Intrusion Detection Based on a General Regression Neural Network Optimized by an Improved Artificial Immune Algorithm. PLoS ONE 10 (3): e0120976. doi:10.1371/journal.pone.0120976

Academic Editor: Peter Csermely, Semmelweis University, HUNGARY

Received: October 14, 2014

Accepted: January 29, 2015

Published: March 25, 2015

Data Availability Statement: All relevant data are within the paper and its Supporting Information files.

Funding: The authors was sponsored by National Natural Science Foundation of China (Grant No: 51309068) and Fundamental Research Funds for the Central Universities. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

Abstract

To effectively and accurately detect and classify network intrusion data, this paper introduces a general regression neural network (GRNN) based on the artificial immune algorithm with elitist strategies (AIAE). The elitist archive and elitist crossover were combined with the artificial immune algorithm (AIA) to produce the AIAE-GRNN algorithm, with the aim of improving its adaptivity and accuracy. In this paper, the mean square errors (MSEs) were considered the affinity function. The AIAE was used to optimize the smooth factors of the GRNN; then, the optimal smooth factor was solved and substituted into the trained GRNN. Thus, the intrusive data were classified. The paper selected a GRNN that was separately optimized using a genetic algorithm (GA), particle swarm optimization (PSO), and fuzzy C-mean clustering (FCM) to enable a comparison of these approaches. As shown in the results, the AIAE-GRNN achieves a higher classification accuracy than PSO-GRNN, but the running time of AIAE-GRNN is long, which was proved first. FCM and GA-GRNN were eliminated because of their deficiencies in terms of accuracy and convergence. To improve the running speed, the paper adopted principal component analysis (PCA) to reduce the dimensions of the intrusive data. With the reduction in dimensionality, the PCA-AIAE-GRNN decreases in accuracy less and has better convergence than the PCA-PSO-GRNN, and the running speed of the PCA-AIAE-GRNN was relatively improved. The experimental results show that the AIAE-GRNN has a higher robustness and accuracy than the other algorithms considered and can thus be used to classify the intrusive data.

Introduction

Network security has become increasingly significant with the development of the Internet. Because of the diversity and complexity of intrusion methods, it is difficult to effectively detect the traditional firewall technique. Network intrusion detection, such as neural networks, appeared at a historic moment [1–3]. The essence of intrusion detection is a classifying problem in which the data of several keys of the Internet or systems are analyzed, the data are classified as normal or abnormal parts using the clustering abilities of neural networks, and a decision is made as to whether the security policy has been broken. Early methods of intrusion detection include techniques based on probabilistic and statistical analyses [4–6] and expert systems in intrusion detection [7, 8]. For example, Forrest S. et al. proposed an intrusion detection method based on comparing the statistical characteristics of the sequences that were called by the systems between the normal behaviors and abnormal behaviors [5]; Warrender C. et al. built a hidden Markov model for the running status of computer systems [6]; Ilgun K. et al. proposed STAT/NSTAT (state transition analysis tool/netstat) based on expert experience to detect the intrusion data [8]; Lindqvist U. et al. introduced an intrusion detection method called P-BEST (production-based expert system toolset) based on a set of rules [7]. Compared with these traditional methods, the applications of neural networks in intrusion detection have many merits. Although methods based on probabilistic and statistical analyses are more mature, they are insensitive to the orders of productions of the intrusions, and the threshold value cannot be accurately ensured. In contrast, because the intrusion activities have more characteristics, the number of their dimensionalities is typically more than 30, and the performance of the traditional statistical analyses will be worse in processing high-dimension data, which causes a high error rate in classifying the intrusion data [9, 10]. Although the approaches based on expert systems have high accuracy, the cost of establishing and maintaining the rule bases is high [4]. Thus, the current main studies on building the structure of expert systems are machine learning and data mining. The neural network is an important part of artificial intelligence and data mining, where back propagation (BP) and radial basis function (RBF) neural networks are the most widely applied. BP neural networks are a type of multi-layer perceptron artificial neural network that can handle a complex non-linear problem, which consists of an input layer, hidden layer, and output layer. However, it has some intrinsic defects; namely, the performance is highly related to the topology structure (the number of hidden layer nodal points) and initial parameters (including the initial connection weights and thresholds of connection nodes), and it easily falls into a local optimal solution [11]. The RBF neural network is similar to the BP neural network. Both networks approach the error by adjusting the weights of neurons [12]. However, the RBF neural network differs from the BP network in some respects. The number of nodes in the hidden layer of the BP network cannot be easily identified, whereas the RBF network can adjust the number of nodes in its hidden layer according to the specific problems; thus, its adaptability is better. Nevertheless, the RBF and BP networks unfortunately have the same defects: low convergence rate and easily falling into a local optimal solution [13]. The RBF network is typically applied in nonlinear time series prediction instead of data classification [13–16]. In contrast, the general regression neural network (GRNN) performs well in nonlinear mapping and has an excellent approximate capability; thus, it overcomes the defects of the traditional BP and RBF networks, such as easily falling into the local optimal solutions and low convergence speed, which makes it suitable for solving the problems of nonlinear functions [13]. Because of these characteristics, the GRNN is often used for data regression and classification and has achieved good results [17–20].

The traditional intelligent algorithms of data regression and classification, such as the traditional GRNN [21], fuzzy C-means clustering (FCM) [22], the GA-GRNN algorithm [23], which combines the genetic algorithm (GA) and GRNN, and the PSO-GRNN algorithm [19], which combines particle swarm optimization (PSO) and the GRNN, have some deficiencies. For example, artificial factors will negatively affect the accuracy of the traditional GRNN because the smooth factors that affect its accuracy are difficult to confirm using experiences or trials. FCM becomes easily trapped in local optimum solutions, and it is difficult to classify [22] because there are many dimensions of the network intrusion characters and the different data categories are similar. In contrast, the performance of FCM and the classification result are affected by the weighting exponent of the FCM, and the traditional FCM cannot guarantee that it obtains the optimal values of the weighting exponent [24]. Thus, the classification may not have the highest accuracy. The GA-GRNN and PSO-GRNN have the same deficiencies as FCM because the data dimensions are high and they have bad local search abilities, which result in premature convergence [25, 26]. The artificial immune algorithm (AIA) uses the diverse generation and maintenance mechanism of immune systems to maintain diverse optimal solutions, and thus, it overcomes the premature convergence of the general optimization processes, particularly the multi-peak functions, and allows global optimum solutions to be obtained. These characters give the AIA higher robustness in solving complex problems [27, 28]. The AIA has been applied in the data-mining field [29]. However, the AIA also has some deficiencies, such as a slow running speed and convergence rate [30]. To overcome these deficiencies, the artificial immune algorithm with elitist strategies (AIAE) was introduced [31]. The elitist archive and elitist crossover were added to the traditional AIA, which improved the adaptivity and accuracy of the algorithm [32, 33].

This study was devoted to exploring a new combinatorial neural network to rapidly determine the intrusion types. Both GRNN and AIAE perform better than their related algorithms. Their combination may produce better results and improve the efficiency of intrusion detection. In addition, the combination of the AIA and neural networks is an important part of algorithm studies [34]. Currently, research on the combination of these two algorithms remains sparse. This study combined AIAE and GRNN and used the excellent global convergence ability of the AIAE to optimize the smooth factors of the GRNN. The optimal smooth factor consists of the optimal structure of the GRNN. Thus, the network intrusion data can be accurately classified, and a better intrusion detection method is obtained.

Materials and Methods

General regression neural network (GRNN)

GRNN is a radial basis function neural network that is composed of an input layer, pattern layer, summation layer and output layer. Its theoretical basis is the nonlinear regression analysis. The essence of the regression analysis, which is composed of the independent variable Y and dependent variable$x$, is to calculate$y$, which has the maximum probability values. Assuming that $f(x, y)$ is the joint probability density function of the random variables x and$y$, the observed value of x is$X$. Thus, the regression of y relative to X is

$$\hat{Y} = E(y|x) = \frac{\int_{-\infty}^{+\infty} yf(x, y)dy}{\int_{-\infty}^{+\infty} f(x, y)dy} \tag{1}$$

where $\hat{Y}$ is the forecast output of Y when the input is$X$.

The density function $\hat{f}(X, y)$ can be estimated based on the sample data set $\{x_i, y_i\}_{i=1}^{n}$ using Parzen non-parametric estimation:

$$\hat{f}(X, y) = \frac{1}{n(2\pi)^{\frac{p+1}{2}}\sigma^{p+1}} \sum_{i=1}^{n} \exp[-\frac{(X - X_i)^T(X - X_i)}{2\sigma^2}]\exp[-\frac{(X - Y_i)^2}{2\sigma^2}] \tag{2}$$

where X_i and Y_i are the observed values of the random variables x and$y$; n is the sample size; p

is the dimension of$x$; and σ is the width coefficient of the Gaussian function, which is called the smooth factor.

$f(x,y)$ is substituted by $\hat{f}(X,y)$ in Eq. 1, and the orders of the integrals and additions are swapped:

$$\hat{Y}(X)=\frac{\sum_{i=1}^{n}\exp[-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2}]\int_{-\infty}^{+\infty}y\exp[-\frac{(Y-Y_i)^2}{2\sigma^2}]dy}{\sum_{i=1}^{n}\exp[-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2}]\int_{-\infty}^{+\infty}\exp[-\frac{(Y-Y_i)^2}{2\sigma^2}]dy} \tag{3}$$

Because $\int_{-\infty}^{+\infty} ze^{-z^2}dz=0$, the output $\hat{Y}(X)$ of the network after the two integrals are calculated is

$$\hat{Y}(X)=\frac{\sum_{i=1}^{n}Y_i\exp[-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2}]}{\sum_{i=1}^{n}\exp[-\frac{(X-X_i)^T(X-X_i)}{2\sigma^2}]} \tag{4}$$

$\hat{Y}(X)$ is the weighted average of all sample observed valuesY_i. Each weight factor of Y_i is the index of the squared Euclidean distance between X_i and$X$. When σ is notably large, $\hat{Y}(X)$ is similar to the average of the dependent variables of all samples. In contrast, when σ approaches 0, $\hat{Y}(X)$ is highly similar to the training samples. If the predicted points are contained in the training sample set, the predictions are highly similar to the corresponding dependent variables in the samples. However, if the points are not contained in the samples, the predictive effect is notably bad, which demonstrates the poor generalization capability of the network. Thus, the optimization of σ directly determines the predictive accuracy of the GRNN [13, 21].

Immune algorithm with elitist strategies

The AIA and GA have similar structures: they consist of the operations of selection, crossover and mutation. Unlike the GA, the quality of individuals (antibodies) is evaluated using the affinity (fitness) and concentration in the AIA, which reflects the diversities of the real immune systems. However, the only evaluation index of the GA is the fitness [35]. Thus, the AIA evaluates the individuals more comprehensively than the GA. Hajiaghaei-Keshteli M defined the parameters of the AIA, which included the antibodies' similarity, concentration, expected breed rate, selection probability and diversity indices [35].

This study used the elitist-archive and elitist-crossover strategies to prevent the loss of the current optimal individuals in the next generation of the population.

Definition 1: In the elitist-archive strategy, the individuals with the maximum fitness, which are called the elitists, must not perform the selection and mutation operations when the experiment begins. The elitists will survive with the selective probability of 100% in the next generation, and they will not be damaged by the selection, crossover and mutation operations. To maintain the scale of the population, the individuals with the minimum fitness will be eliminated if the elitists are added into the next generation [32].

Definition 2: In the elitist-crossover strategy, the elitists should be crossed with the chosen individuals in the population according to a pre-set elitist-crossover probability in addition to the traditional crossover; thus, the genetic structure of the population can be improved. The theory supporting this strategy is provided in Tan GZ et al. [33]. The number of excellent patterns can be increased using the elitist crossover. However, the diversity of the population will decrease if the algorithm only adopts the elitist crossover. Thus, the traditional crossover should be used in combination with the elitist crossover so that it maintains the population diversity and improves the total population quality.

Principal component analysis

Principal component analysis (PCA) is a type of technology that can analyze and simplify data sets. The dimensions of the data sets can be decreased by reserving the low-order principal components and neglecting the high-order ones; simultaneously, the characteristics with the largest contribution for the variances in the data sets are reserved. Such low-order components can reserve the most important aspects of data. However, because the PCA is extremely sensitive to extrema and missing values, these factors may produce missing or incorrect results [36, 37]. Thus, it will negatively affect the study of neural networks and the classification accuracy.

Methods

This paper introduces an AIAE-GRNN algorithm to improve the detection accuracy. The mean square errors (MSEs) of the network model created by the GRNN were taken as the affinity functions. The smooth factors were optimized by iterating the AIAE until the algorithm satisfied the terminal condition. The smooth factor that was solved at this time is the optimal one. Then, the GRNN with the optimal smooth factor was used to classify the intrusion data. The algorithm flow chart is shown in Fig. 1. The concrete steps are:

Step 1: Data preprocessing. To eliminate the magnitude differences among the data of each dimension and improve the classification accuracy, the intrusion data could be normalized into the figures in [−1, 1] [38]. Because there were 38 indices in the input section of the intrusion data, the PCA could be used in the dimensionality reduction. However, the accuracy may decline because the PCA only extracts characteristics with the largest contributions to the variances in the data and neglects some secondary characteristics. This problem will be discussed later.

Step 2: Antigen recognition. The affinity functions, which were the antigens in the immune system, were set as the MSEs of the output values and actual values in each generation of the GRNN. The initial parameters of the antigens, such as the population size N, crossover probability P_c, elitist-crossover probability P_{ck} and mutation probability P_v, were set. The running parameters of the algorithm, such as the maximum iterations and capacity of the memory bank, were also set.

Step 3: Production of the initial antibodies. N smooth factors were randomly produced as the initial solutions.

Step 4: Calculation of the antibody affinities. The intrusive data were plugged into the GRNN after normalization; then, the output values of the MSE in this generation were produced.

Step 5: Elitist archive. The current individuals with the largest affinities were preserved in the memory bank as memory cells in a certain proportion. Each antibody affinity was

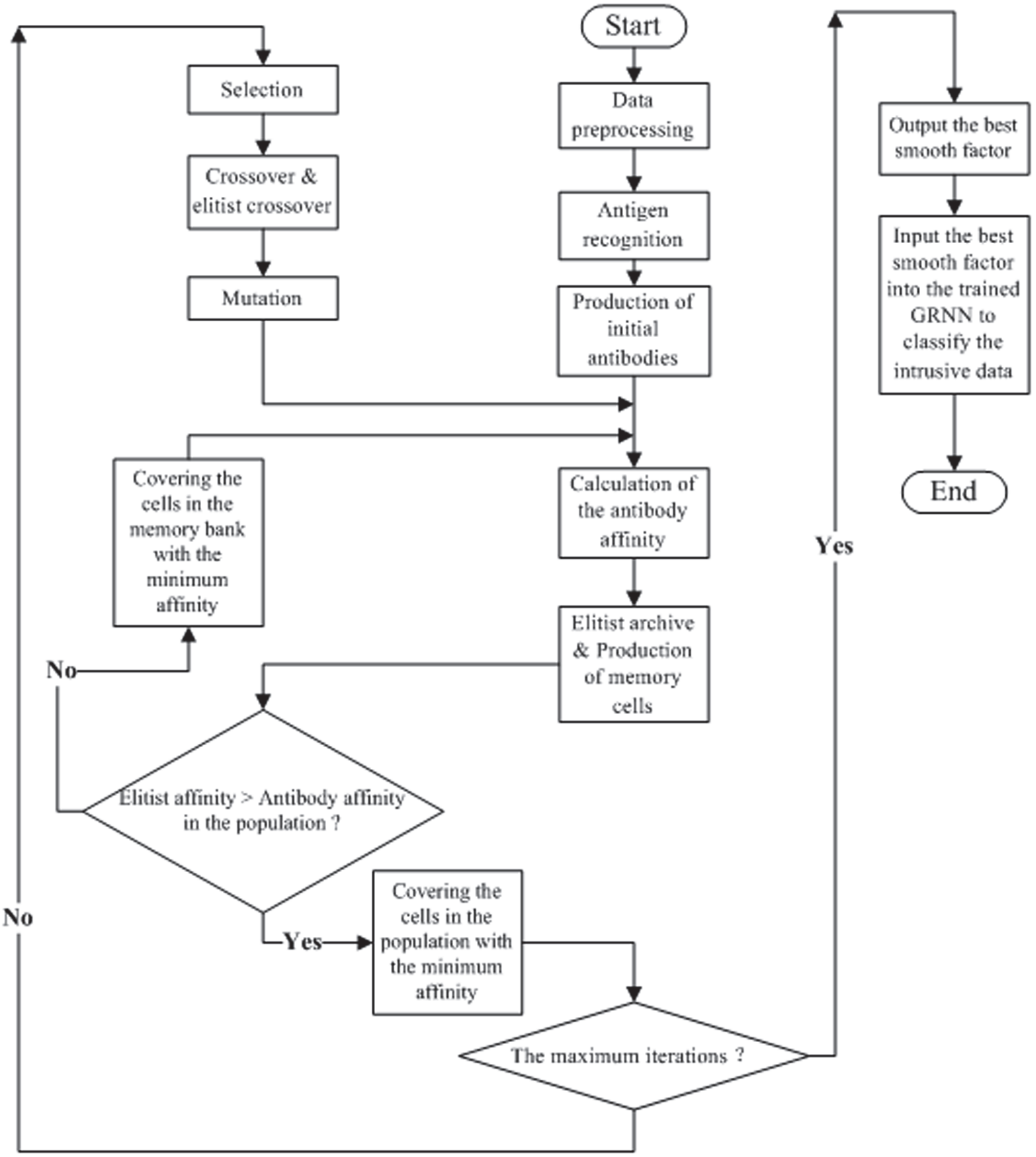
Start
Data preprocessing
Antigen recognition
Production of initial antibodies
Calculation of the antibody affinity
Elitist archive & Production of memory cells
Elitist affinity > Antibody affinity in the population ?
Yes
No
Covering the cells in the population with the minimum affinity
Covering the cells in the memory bank with the minimum affinity
The maximum iterations ?
No
Yes
Selection
Crossover & elitist crossover
Mutation
Output the best smooth factor
Input the best smooth factor into the trained GRNN to classify the intrusive data
End

Fig 1. The algorithm flow chart of AIAE-GRNN.

doi:10.1371/journal.pone.0120976.g001

calculated, and if all current antibodies had lower affinities than elitists, the antibodies with the lowest affinities would be replaced by the elitists. Then, the algorithm continued to Step 6. In contrast, if some antibodies had higher affinities than the elitists, these antibodies would replace the elitists with the lowest affinities in the population; then, the algorithm would repeat Step 4.

Step 6: Judgment of the loop's termination condition. If the algorithm achieved the maximum iterations, it would be terminated and output the current smooth factor. Then, the smooth factor would be an input of the trained GRNN to classify the intrusive data. Otherwise, the algorithm would continue to Step 7.

Step 7: Calculation of the antibody concentration and expected breed rate. Each antibody concentration and expected breed rate would be calculated according to the computing methods of Hajiaghaei-Keshteli M [35].

Step 8: Selection. The algorithm would perform the selection operation. The probability of the individuals under selection was the expected breed rate, which had been calculated.

Step 9: Crossover and elitist crossover. The population would be crossed with probability P_c; then, the elitist-crossover operation would be performed on the individuals in the population using the methods of the previous study [33], and the probability P_{ck} of the operation was notably low.

Step 10: Mutation. The mutative positions, which were randomly selected, would be mutated with probability P_k; then, the affinities of the variant individuals would return to Step 4 to be calculated again.

Simulation and Testing

Data

The experimental environment was based on Microsoft Windows 7. The hardware environment was Intel Core i5-3230M, 2.60 GHz and RAM 4 GB, and the simulation software was MATLAB 2012b (MathWorks, located in Natick, Massachusetts, U.S.A.). The experimental data were divided into the training data (S1 Data) and the testing data (S2 Data). All the data were obtained from the network intrusion data set KDD CUP 1999 [39], which included normal data (Normal) and four categories of abnormal data of a denial-of-service attack (DoS), probing attack (PROBE), remote-to-login attack (R2L) and user-to-root attack (U2R). The input of this data set has 38 indices, and its output has one index, which is the abnormal category. The training and testing samples are shown in S1 Table.

Contrast models and evaluation indices

In addition to the AIAE-GRNN, the study selected two common GRNN combinational algorithms, GA-GRNN and PSO-GRNN, as the controlled trials, and FCM was the control sample. In fact, the convergence of FCM is directly related to the selection of the weighting exponent. The value of the weighting exponent in the traditional FCM must be chosen by experience; thus, the human factors will seriously affect the performance of the FCM [24]. This factor has a less significant effect on convergence compared to other algorithms. Similarly, the traditional GRNN was not selected, because it is identical to the FCM, where the smooth factor may not be the optimal one. Therefore, the FCM was only selected as a referential control, and the convergence of FCM was not discussed. In addition, some papers [40, 41] have proven that the convergence of the traditional FCM is not sufficient, and the traditional FCM typically has to be combined with other algorithms, such as a GA. Thus, FCM is not used by itself, and it is not selected to compare the convergence with other algorithms. The evaluation indices were the detection rate (DR), false positive rate (FPR), maximum iterations and running time [2, 3]. The algorithms were tested after being classified according to the abnormal categories. Before the formal test, a number of preliminary tests were necessary, and reasonable experimental parameters were set according to previously published work [42–44]. Thus, a better experimental result was obtained, and the running time was not excessively large. The maximum number of iterations was 100; the population size N of each algorithm was 30; the crossover probability P_c and mutation probability P_v of the AIAE and GA were 0.5; the two acceleration factors of PSO, i.e., c_1 and c_2, were 2; the AIAE evaluation diversity was 0.95; the capacity of the memory bank

Table 1. DR & FPR of each intrusion category by the different algorithms.

Categories	DoS		R2L		Probe		U2R	
Methods	DR	FPR	DR	FPR	DR	FPR	DR	FPR
GA-GRNN	99.43	0.36	98	0.15	99.06	0.56	98.32	0.22
PSO-GRNN	99.54	0.23	97.59	0.11	99.2	0.49	98.75	0.21
AIAE-GRNN	99.88	0.05	97.98	0.06	99.44	0.36	98.89	0.16
FCM	90.53	2.65	56.48	0.63	82.67	1.96	76.88	0.23

The result showed that the DR and FPR of PSO-GRNN and AIAE-GRNN were higher than GA-GRNN and FCM. And the DR and FPR of AIAE-GRNN were higher slightly than PSO-GRNN.

doi:10.1371/journal.pone.0120976.t001

was 10; and the elitist-crossover probability P_{ck} was 0.05. The definitions of DR and FPR are as follows [45]:

Definition 3: The detection rate (DR) is the probability that the detection systems can correctly give an alarm when the monitoring systems are attacked.

Definition 4: The false positive rate (FPR) is the probability that the normal instructions are mistakenly considered intrusion instructions, and an alarm is given.

Results and further discussion

Table 1 shows the DR and FPR of the algorithms, and Table 2 shows the relationship between the convergence iterations and the running time. Fig. 2A shows the relationship between the optimized GRNN smooth factors and the iterations, and Fig. 2B shows the relationship between the MSE and number of iterations.

Table 1 shows that the AIAE-GRNN has the highest DR, but its running time is the longest because of the high algorithm complexity of the AIAE [30, 46–48]. Although the PSO-GRNN has a slightly lower DR than the AIAE-GRNN, it has the same convergence rate and a faster running speed than the AIAE-GRNN. Their ratio of running time is 0.6576. As shown in Fig. 2A, the GA-GRNN appears to have a premature convergence. During the 78th iteration, the values of the GRNN smooth factors continue to change by a large margin, which shows that it has a worse local search ability than the AIAE-GRNN and PSO-GRNN and a lower classification accuracy than the AIAE-GRNN. Although the traditional FCM has a short running time, its classification accuracy is not comparable with other intelligence algorithms, which

Table 2. The evaluation indexes for the different algorithms.

Methods	GA-GRNN	PSO-GRNN	AIAE-GRNN	FCM
Fitness	0.0141	0.0141	0.0141	—
Sooth factors	24.8551	24.8522	24.8542	—
Convergence generations	89	8	8	—
Running time	19min38s	19min20s	29min24s	8s

The result showed that the GA-GRNN had the premature convergence problem. In contrast, the PSO-GRNN and AIAE-GRNN overcame this problem. The running time of PSO-GRNN was shorter than AIAE-GRNN.

doi:10.1371/journal.pone.0120976.t002

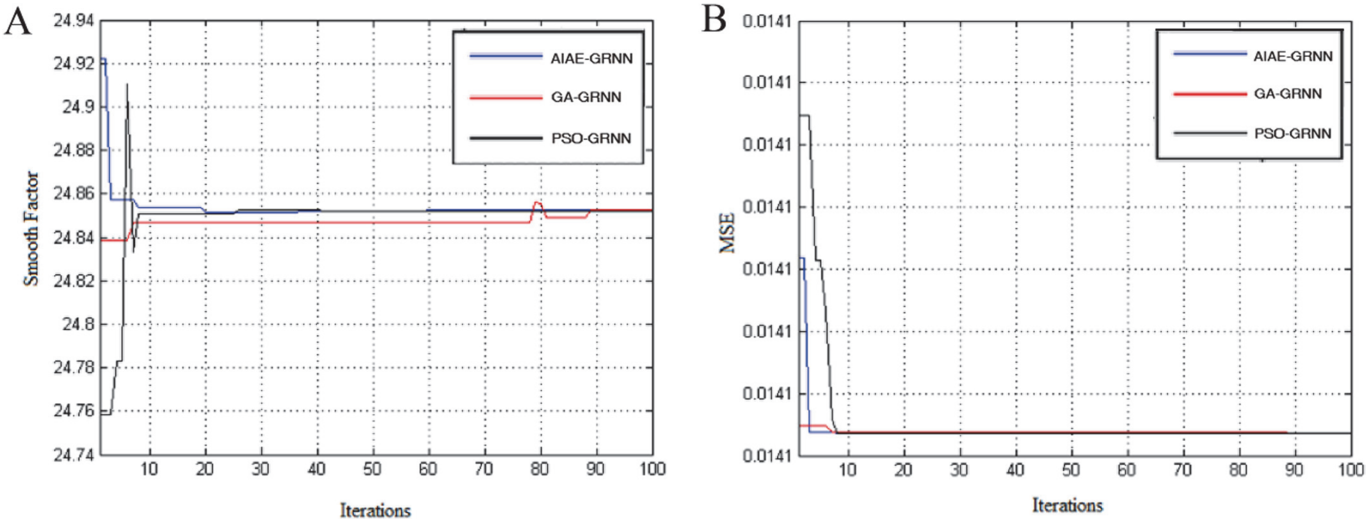

Fig 2. The output performances of GA-GRNN, PSO-GRNN and AIAE-GRNN. (A) The relationship between the optimized GRNN smooth factors and the iterations of GA-GRNN, PSO-GRNN and AIAE-GRNN. (B) The relationship between MSE and the iterations of GA-GRNN, PSO-GRNN and AIAE-GRNN.

doi:10.1371/journal.pone.0120976.g002

results in a difficult classification (Table 1). Thus, both the AIAE-GRNN and PSO-GRNN have merits and defects, and further experiments were necessary to test the performance of these two algorithms. However, the GA-GRNN and FCM were eliminated.

PCA was used to test the two remaining algorithms (AIAE-GRNN and PSO-GRNN). PCA can reduce the dimensions of the training samples and training time [49], but the classification accuracy may be affected. Thus, it was necessary to test this approach using experiments. PCA was performed for the input data using SPSS 19. Fourteen types of input evaluation indices had the largest eigenvalues, with a cumulative contribution of 86.42% determined using the SPSS 19 analysis. When the cumulative contribution is above 85%, the largest eigenvalues can represent the principal characteristics of the entire data set [50]. Thus, these 14 input indices were chosen as the training and testing samples.

The value of the maximum iteration was 30, and other parameters were identical to those in the previous paragraphs. The DR and FPR of the two networks are shown in Table 3, and the convergence iterations and running time are shown in Table 4. The relationship between the optimized GRNN smooth factors and the iterations is shown in Fig. 3A, and the relationship between MSE and the iterations is shown in Fig. 3B.

From Table 2, compared with the network without PCA, the DRs of the 4 intrusion categories that were optimized using PCA-AIAE-GRNN decrease by only 0–0.5%, but all FPRs are below those of the previous AIAE-GRNN. However, the DR of the 4 intrusion categories that

Table 3. DR & FPR of each intrusion category by PCA-PSO-GRNN and PCA-AIAE-GRNN.

Categories	DoS		R2L		Probe		U2R	
Methods	DR	FPR	DR	FPR	DR	FPR	DR	FPR
PCA-PSO-GRNN	97.74	0.35	95.88	0.04	97.5	0.45	98.81	0.15
PCA-AIAE-GRNN	99.41	0.01	98.4	0	99.01	0.22	98.45	0.03

By reducing dimensions in PCA, compared with Table 1, the DR and FPR of PSO-GRNN and AIAE-GRNN declined to a certain extent, but the DR and FPR of AIAE-GRNN was still higher than PSO-GRNN.

doi:10.1371/journal.pone.0120976.t003

Table 4. The evaluation indexes for PCA-PSO-GRNN and PCA-AIAE-GRNN.

Methods	PSO-GRNN	AIAE-GRNN
Fitness	0.0141	0.0141
Sooth factors	23.4547	24.8546
Convergence generations	23	7
Running time	3min58s	6min3s

By reducing dimensions in PCA, compared with Table 2, the convergence and relative running time were improved. This result showed that the robustness of AIAE-GRNN was better than PSO-GRNN.

doi:10.1371/journal.pone.0120976.t004

were optimized using PCA-PSO-GRNN decreases by 1.7–2.1%; in particular, the R2L has the largest decrease range of 2.12%, and only the DR of U2R increases slightly. Their running time ratio is 0.6556, which is lower than the ratio without PCA. This phenomenon indicates that the PCA-AIAE-GRNN has a faster running speed than the previous networks. Fig. 3A & B show that the PCA-PSO-GRNN has worse convergence than the PCA-AIAE-GRNN and appears to have a premature convergence, which indicates a poor local search ability. The results of further experiments show that the AIAE-GRNN has better robustness than the PSO-GRNN, and the accuracy is not strongly affected by the application of PCA.

Conclusion

The study introduced a method of network intrusion detection using the AIAE-GRNN. The AIAE-GRNN has a higher classification accuracy but longer running time than two types of GRNN combinational algorithms and FCM. To decrease the running time and prove the robustness of the network, this study used PCA to reduce the dimensions of the intrusive data, which effectively reduced the training time of the network. In the contrast test, which contained PSO-GRNN that was similarly processed with PCA, the classification accuracy of AIAE-GRNN with the shortened running time is barely affected, and the AIAE-GRNN has better convergence than the PSO-GRNN. This result indicates that the AIAE-GRNN is not

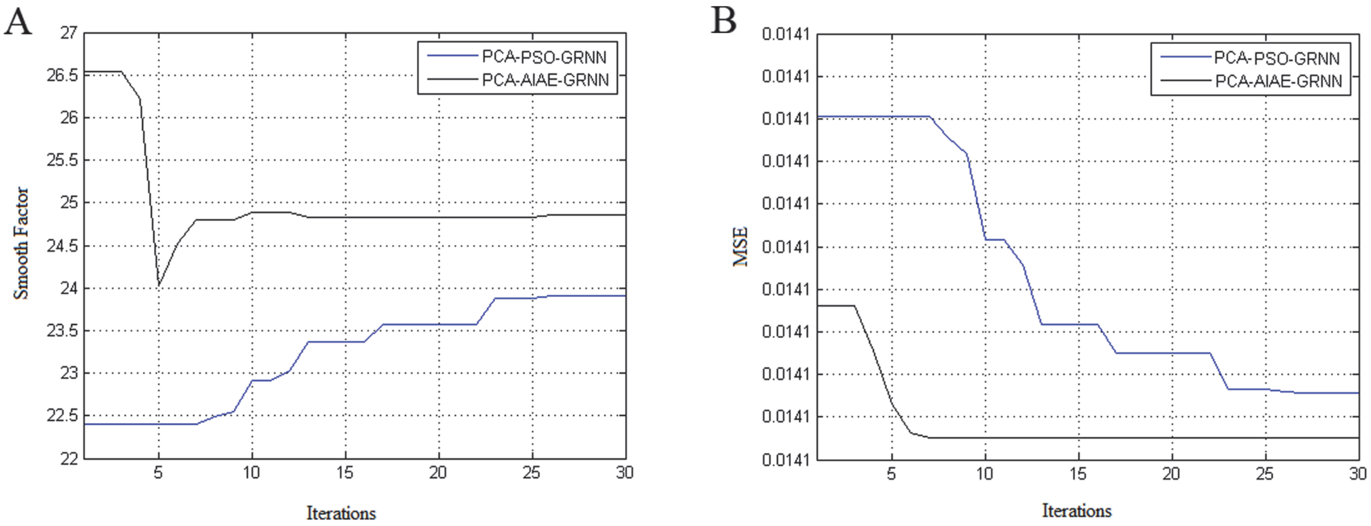

Fig 3. The output performances of PCA-AIAE-GRNN and PCA-PSO-GRNN. (A) The relationship between the optimized GRNN smooth factors and the iterations of PCA-AIAE-GRNN and PCA-PSO-GRNN. (B) The relationship between MSE and the iterations of PCA-AIAE-GRNN and PCA-PSO-GRNN.

doi:10.1371/journal.pone.0120976.g003

notably sensitive for the training samples and shows a good robustness. The PCA-AIAE-GRNN balances the classification accuracy and running efficiency, and can therefore be used in intrusion detection or other fields. Although the AIAE-GRNN has many merits, the AIAE has a lower running speed than PSO and other intelligence algorithms. Optimizing the time complexity of the AIAE-GRNN will be critical to improving the network performance and must be studied further.

Supporting Information

S1 Table. The training and testing samples for the networks.
(DOC)

S1 Dataset. The training data.
(XLS)

S2 Dataset. The testing data.
(XLS)

Acknowledgments

The authors would like to thank Mr. Ming-Hao Zhao for language assistance.

Author Contributions

Conceived and designed the experiments: JW HL. Performed the experiments: JW DP. Analyzed the data: JW DP LZ. Contributed reagents/materials/analysis tools: ZL LZ. Wrote the paper: JW ZL. Designed the algorithm and experiment methods: JW HL. Programming and data compilation: DP. Wrote the thesis: JW ZL.

References

1. Tian D, Liu YH, Xiang Y. Large-scale network intrusion detection based on distributed learning algorithm. Int J Inf Secur. 2009; 8: 25–35.
2. Beghdad R. Critical study of neural networks in detecting intrusions. Comput Secur. 2009; 27: 168–175.
3. Tong XJ, Wang Z, Yu HN. A research using hybrid RBF/Elman neural networks for intrusion detection system secure model. Comput Phys Commun. 2009; 180: 1795–1801.
4. Wei YX, Wu MQ. Intelligent grid intrusion detection system. J Softw (in Chinese). 2006; 17: 2384–2394.
5. Forrest S, Hofmeyr SA, Somayaji A, Longstaff TA. A sense of self for unix processes. In: Security and Privacy, 1996. Proceedings. 1996 Symposium on. IEEE. 1996;pp. 120–128.
6. Warrender C, Forrest S, Pearlmutter B. Detecting intrusions using system calls: Alternative data models. In: Security and Privacy, 1999. Proceedings. 1999 Symposium on. IEEE. 1999;pp. 133–145.
7. Lindqvist U, Porras PA. Detecting computer and network misuse through the production-based expert system toolset (P-BEST) In: Security and Privacy, 1999. Proceedings. 1999 Symposium on. IEEE. 1999;pp. 146–161.
8. Ilgun K, Kemmerer RA, Porras PA. State transition analysis: A rule-based intrusion detection approach. IEEE T Software Eng. 1995; 21: 181–199.
9. Hua J, Tembe WD, Dougherty ER. Performance of feature-selection methods in the classification of high-dimension data. Pattern Recogn. 2009; 42: 409–424. doi: 10.1002/eat.20644 PMID: 19115362
10. Liu Y, Hayes DN, Nobel A, Marron JS. Statistical significance of clustering for high-dimension, low–sample size data. J Am Stat Assoc. 2008; 103: 1281–1293.
11. Wang J, Sheng Z, Zhou B, Zhou S. Lightning potential forecast over Nanjing with denoised sounding-derived indices based on SSA and CS-BP neural network. Atmos Res. 2014; 137: 245–256.
12. Yang J, Zhou Y, Zhou J, Chen Y. Prediction of Bridge Monitoring Information Chaotic Using Time Series Theory by Multi-step BP and RBF Neural Networks. Intell Autom Soft Co. 2013; 19: 305–314.

13. Cigizoglu HK, Alp M. Generalized regression neural network in modelling river sediment yield. Adv Eng Softw. 2006; 37: 63–68.
14. Lee CM, Ko CN. Time series prediction using RBF neural networks with a nonlinear time-varying evolution PSO algorithm. Neurocomputing. 2009; 73: 449–460. doi: 10.1016/j.urology.2008.08.495 PMID: 19185765
15. Rank E. Application of Bayesian trained RBF networks to nonlinear time-series modeling. Signal Process. 2003; 83: 1393–1410.
16. Rojas I, Pomares H, Bernier JL, Ortega J, Pino B, Pelayo FJ, et al. Time series analysis using normalized PG-RBF network with regression weights. Neurocomputing. 2002; 42: 267–285.
17. Yap KS, Lim CP, Abidin IZ. A hybrid ART-GRNN online learning neural network with a epsilon-insensitive loss function. IEEE T Neural Networ. 2008; 19: 1641–1646. doi: 10.1109/TNN.2008.2000992 PMID: 18779094
18. Kim SW, Singh VP, Seo YM, Kim HS. Modeling nonlinear monthly evapotranspiration using soft computing and data reconstruction techniques. Water Resour Manag. 2014; 28: 185–206.
19. Lin SM. Analysis of service satisfaction in web auction logistics service using a combination of Fruit fly optimization algorithm and general regression neural network. Neural Comput Appl. 2013; 22: 783–791.
20. Erkmen B, Yıldırım T. Improving classification performance of sonar targets by applying general regression neural network with PCA. Expert Syst Appl. 2008; 35: 472–475.
21. Choudhary AK, Swarup A. Performance of intrusion detection system using GRNN. Int J Comput Sci Netw Secur. 2009; 9: 219–222.
22. Jawhar MMT, Mehrotra M. A hybrid FCM clustering- neural network model for intrusion detection. Adv Mater Res. 2012; 403–408: 3519–3527.
23. Yao SB, Guo DL, Yang GW. Three-dimensional aerodynamic optimization design of high-speed train nose based on GA-GRNN. Sci China Technol Sc. 2012; 55: 3118–3130.
24. Yu J, Cheng Q, Huang H. Analysis of the weighting exponent in the FCM. IEEE T Syst Man Cy B. 2004; 34: 634–639. PMID: 15369100
25. Choubey NS, Kharat MU. Hybrid system for handling premature convergence in GA—Case of grammar induction. Appl Soft Comput. 2013; 13: 2923–2931.
26. Lu HY, Sriyanyong P, Song YH, Dillon T. Experimental study of a new hybrid PSO with mutation for economic dispatch with non-smooth cost function. Int J Elec Power. 2010; 32: 921–935.
27. Timmis J, Hone A, Stibor T, Clark E. Theoretical advances in artificial immune systems. Theor Comput Sci. 2008; 403: 11–32.
28. Chen JY, Yang DY. Data security strategy based on artificial immune algorithm for cloud computing. Appl Math Inform Sci. 2013; 7: 149–153.
29. Freitas AA, Timmis J. Revisiting the foundations of artificial immune systems for data mining. IEEE T Evolut Comput. 2007; 11: 521–540.
30. Ge H. The applications of immune algorithm and kernel clustering artificial immune network. D.Eng. Thesis, College of Automation Science & Engineering, South China University of Technology. 1999.
31. Wang Q, Wang RB, Chen J. Design and application of PID controller based improved artificial immune algorithm. In: Systems and Control in Aerospace and Astronautics, 2008 (ISSCAA 2008). 2nd International Symposium on. 2008;pp. 1–5.
32. Wang YN, Wu LH, Yuan XF. Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure. Soft Comput. 2010; 14: 193–209.
33. Tan GZ, Zhou DM, Jiang B. Elitism-based immune genetic algorithm and its application to optimization of complex multi-modal functions. J Cent South Univ T. 2008; 15: 845–852.
34. Dasgupta D, Yu S, Nino F. Recent advances in artificial immune systems: models and applications. Appl Soft Comput. 2011; 11: 1574–1587.
35. Hajiaghaei-Keshteli M. The allocation of customers to potential distribution centers in supply chain networks: GA and AIA approaches. Appl Soft Comput. 2011; 11: 2069–2078.
36. Sarbu C, Pop HF. Principal component analysis versus fuzzy principal component analysis—A case study: the quality of danube water (1985–1996). TALANTA. 2005; 65: 1215–1220. doi: 10.1016/j.talanta.2004.08.047 PMID: 18969934
37. Marengo E, Robotti E, Righetti PG, Antonucci F. New approach based on fuzzy logic and principal component analysis for the classification of two-dimensional maps in health and disease. Application to lymphomas. J Chromatogr A. 2003; 1004: 13–28. PMID: 12929957

38. Cao JC, Cao SH. Study of forecasting solar irradiance using neural networks with preprocessing sample data by wavelet analysis. Energy. 2006; 31: 3435–3445.

39. KDD CUP 1999. UCI Machine Learning Repository. 1999. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. Accessed 15 April 2014.

40. Gao F, Ru A, Wang J, Mao S. Knowledge-based detection method for SAR targets. J Syst Eng Electron. 2014; 25573–579.

41. Guo HX, Liao XW, Zhu KJ, Ding C, Gao YH. Optimizing reservoir features in oil exploration management based on fusion of soft computing. Appl Soft Comput. 2011; 11: 1144–1155.

42. Zheng RR, Mao ZY, Luo XX. Artificial immune algorithm based on euclidean distance and king-crossover. Control Decis (in Chinese). 2005; 20: 161–164, 169.

43. Liang Y, Yu HB, Zeng P. Optimization of cluster-based routing protocols in wireless sensor network using PSO. Control Decis (in Chinese). 2006; 21: 453–456, 461.

44. Gao W. An improved fast-convergent genetic algorithm and its performance study. Syst Eng Electron (in Chinese). 2003; 25: 1427–1430.

45. Lippmann R, Haines JW, Fried DJ, Korba J, Das K. The 1999 DARPA off-line intrusion detection evaluation. Comput Netw. 2000; 34: 579–595.

46. He J, Yao X. Drift analysis and average time complexity of evolutionary algorithms. Artif Intell. 2001; 127:57–85.

47. He J, Yao X. A study of drift analysis for estimating computation time of evolutionary algorithms. Nat Comput. 2004; 3: 21–35.

48. Ye HT, Luo F, Xu YG, Yang H, Liang J. Optimal control of wastewater treatment process based on novel immune algorithm. J South China Univ T (Nat Sci Edit) (in Chinese). 2010; 2: 28.

49. Xiao YQ, He YG. A novel approach for analog fault diagnosis based on neural networks and improved kernel PCA. Neurocomputing. 2011; 74: 1102–1115.

50. Liu XD, Jiang BF, Gu WD, Liu QY. Temporal trend and climate factors of hemorrhagic fever with renal syndrome epidemic in Shenyang City, China. BMC Infect Dis. 2011; 11: 331. doi: 10.1186/1471-2334-11-331 PMID: 22133347