Science Arena Publications
Specialty Journal of Electronic and Computer Sciences
ISSN: 2412-7485
Available online at www.sciarena.com
2019, Vol, 5 (1): 58-66

Prediction of Content Error in Cloud Computing based on Perceptron Neural Network and Radial Basis Function (RBF)

Reza Imani

Department of Information Technologhy, Faculty of Engineering, Tabriz Branch, Islamic Azad University, Bonab, Iran.

Abstract: *Cloud computing is a general term for referring to anything that requires the provision of services hosted on the Internet. With advance in cloud computing, data error prediction has become an important factor in cloud computing, so that predicting cloud errors is the most important barrier to the speed and development of cloud computing software. The purpose of the study was to predict content error in cloud computing based on perceptron neural network and RBF. In this research, a data set of 10,000 records has been used, including 23 features that were generally divided into three categories, properties related to public security and properties related to the content of the data and the characteristics required for cloud storage. In this research, the KFOLD method was selected as the allocation of training and testing the data. The method of data selection was random. The software used in this research was MATLAB, 2015. The results showed that the performance of the RBF system was better than the other method in the two predictive systems.*

Keywords: *Cloud computing, Perceptron neural network, RBF.*

INTRODUCTION

The gradual change in the significance of establishing communication between the two systems on the importance of content distribution and the current Internet architecture, which is based on the link between the two systems, is no longer able to meet the growing needs of users and organizations. Thus, a new architecture is proposed for providing content access based on the name of that content. This is called cloud computing system. This technology is a growing trend in information technology presenting low-cost, dynamic computing solutions. Hence, many improvements have been proposed in this regard to enhance the prediction of error and privatization in cloud computing, eliminating the worries, and preventing the users from benefiting from the benefits of cloud computing (Soule, Salamatian and Taft, 2005).

Cloud computing is a general term to refer to anything in need of the provision of services hosted on the Internet. Cloud computing in the organizations trying to reduce their Information Technology (IT) costs by transferring their software costs to third party organizations providing services such as software as a service and platform as a service and infrastructure as a service has attracted much attention (Barford et al., 2002).

Cloud computing means developing and using computer technology based on the Internet; i.e., computer computing takes place in a space where IT-related capabilities are offered as services for the user that allows him to access technology-based services on the Internet - without having specialized information about these technologies or taking control of the technology infrastructure that supports them. The services presented in

the cloud provide online applications accessible to the web browser. At this time, software and data are stored on servers (Ghaffari. 2010).

Error prediction concerns like sending malicious files, stealing information, manipulating data by unauthorized people and so on as well as privacy concerns due to the user's hosting of several places, lack of deployment of information in specific locations, the provider of the service, and so on are the cases that result in ad results due to the specific features of this technology in case of non-use of error prediction mechanisms (Lakhina, Crovella and Diot, 2004). The cloud-computing environment is a dynamic environment, so it needs its special error prediction strategies. Thus, the problem of encryption of information in these networks is not examined well (Lakhina, Crovella and Diot, 2004).

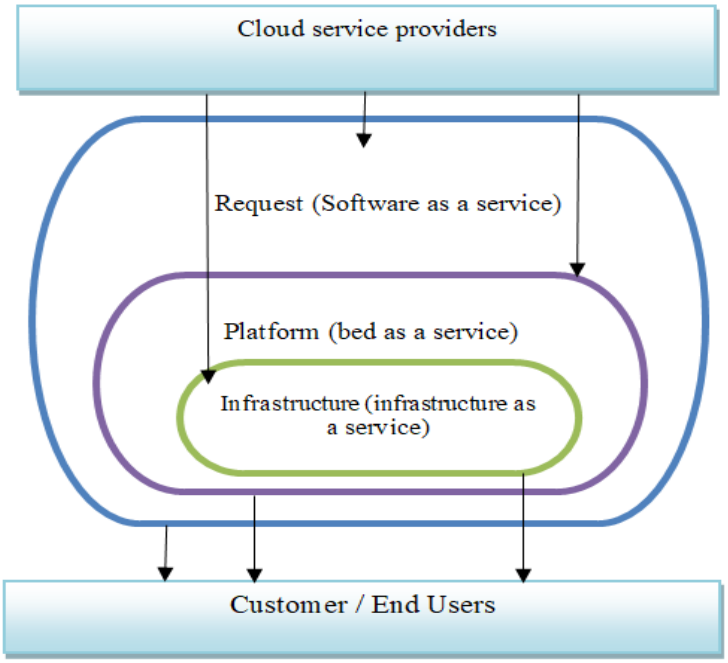


Figure 1: Structural architecture of cloud computing (Munir and Palaniappan, 2013)

Some studies have been conducted in this field. In (Lo, Huang and Ku, 2010), a mold cloud was designed to detect system intrusion, where each machine was considered as a separate block and three separate modules were considered for its intrusion detection system (IDS). These three modules were block, communication, and cooperation. The cooperation part used messages to coordinate with other parts. By exchanging information between their IDSs, they voted for each other's messages, and finally, a coordinated strategy was adopted against the attack. Then each IDS added a command block to its table based on the designated strategy to react to this type of attack.

In (Roschke, Cheng and Meinel, 2009), an IDS has been designed, which was in line with the concept of virtualization and thus, worked efficiently. However, the problem with this system was that users could order and decide on the system as an operator. Hence, the user may not recall the proper command and cause further problems due to the lack of awareness about the attack.

In (Lee et al., 2011), IDSs and multi-level input management were designed in cloud computing. The problem with this system was providing these rules and determining the risks associated with them. Even in some cases, one cannot use all cases for all the users or assign similar risks to them. Moreover, the provision of these rules needs the knowledge of the field of application of the user and their applications, which ends in heavy and complex, and sometimes time-consuming processing. However, it has its own advantages: it deals

more with intra-network discussions, especially resources, and avoids processing on external factors, which means saving on redundant processing. Nonetheless, this causes a fault in the system. Placement of IDS has been considered in a cloud-computing environment in (Mazzariello, Bifulco and Canonico, 2010). Among the disadvantages of this, is the lengthy and costly detection process and that the controller of its parts can be a bottleneck, slowing down the system and even stop it. Accordingly, the purpose of the study was investigating whether using Perceptron Neural Network and RBF methods in cloud computing would lead to the prediction of content error.

Methods

A dataset of 10,000 records was used in the study, including 23 features generally divided into three categories: features related to public security and features related to data content and the characteristics needed for cloud storage. In this research, KFOLD method was selected as assigning training and testing the data. The method of data selection was random. The software used was MATLAB, 2015. Criteria for evaluating the quality of error predictive models in software include:

1. The rate class wrong classification

The most commonly used scale to evaluate the performance of predictive error models is MR, which is the ratio of the number of false module classifications to the total number of modules. MR confusion matrix can be obtained according to the following equation:

$$MR = \frac{FP + FN}{TP + TN + FP + FN} \qquad (1)$$

2. The cost of false miscalculation

ECM is a criterion for comparing the performance of different software quality categorization models. There is a function of cost of classification error based on classification error (Err |) (an npf module is classified as fp) and the classification error type || (Err ||) (a module fp is classified as npf) that ECM is used to calculate the ratio of these different costs.

Evaluation of software quality modeling is very important in presenting a difference in cost ratios, as the utility of a model depends on the cost of its false classification. Errors Err| and Err|| can be obtained from confusion errors` matrix:

$$Err_{|} = \frac{FP}{TN + FP} \qquad (2)$$

$$Err_{||} = \frac{FN}{TP + FN} \qquad (3)$$

As the cost of these errors is different and a single measure for cost is needed, ECM was used and calculated according to the following equation:

$$ECM = C_{|}Err_{|}P_{ndf} + C_{||}Err_{||}P_{df} \qquad (4)$$

$C_{|}$ and $C_{||}$ are the costs of the errors $Err_{|}$ and $Err_{||}$, respectively, and P_{ndf} and P_{df} are the probabilities of non-error-prone and error-prone modules, respectively.

3. Normalized value of false calculation

In many cases, one cannot obtain the cost of any false classifications separately, so normalized EMC (NEMC) is used.

$$NECM = Err_{|}P_{ndf} + \frac{C_{||}}{C_{|}} Err_{||}P_{df} \qquad (5)$$

$C_{|}$ and $C_{||}$ are the costs of the errors $Err_{|}$ and $Err_{||}$, respectively, and P_{ndf} and P_{df} are the probabilities of non-error-prone and error-prone modules, respectively.

4. Sensitivity

This criterion shows the precision of the prediction model and is defined as the percentage of classes that are correctly predicted to be error-prone.

$$\text{Sensitivity} = \frac{Number\ of\ moduls\ correctly\ predicted\ as\ fault\ prone}{total\ number\ of\ actual\ \ faulty\ modules} \times 100 \qquad (6)$$

$$Err_{||} = \frac{FN}{TP + FN}$$

5. Specificity

This criterion, like sensitivity, shows the accuracy of the prediction model, defined as the percentage of classes that are correctly non-fault prone for predicting the error.

$$Sensitivity = \frac{Number\ of\ moduls\ correctly\ predicted\ as\ non-fault\ prone}{total\ number\ of\ actual\ non\ faulty\ modules} \times 100 \qquad (7)$$

6. Accuracy

Accuracy is defined as the number of classes correctly predicted (with and without errors) to the total number of classes. Accuracy (or success rate) is used to measure the overall accuracy of prediction precision as defined by the following equation:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \qquad (8)$$

7. Precision

Precision is defined as the number of error-prone classes that are correctly predicted by the model as susceptible to error. Its best value is 1. More precision means less FP (error-free elements are mistakenly classified as error-prone):

$$\text{Pr}\,ecision = \frac{TP}{TP + FP} \qquad (9)$$

8. Recalling

Recalling is the number of probable error classes predicted by the model as the prediction of error. Its best value is 1. High recalling means low FN.

$$\text{Re}\,call = \frac{TP}{TP + FN} \qquad (10)$$

9. Combined Factor (F)

Combined Factor (F) considers both accuracy and recall for precision, which can be interpreted as a weighted average of accuracy and recall. This weight is shown by α and is usually considered to be 1. F value ranges from 0 and 1, and as it is closer to 1, it has better performance for classification results.

$$F-measure=\frac{2*\mathrm{Re}\,call*\mathrm{Pr}\,ecision}{\mathrm{Re}\,call*\mathrm{Pr}\,ecision} \quad (11)$$

10. Consistency

Increasing the consistency makes the model even more precise. If TP = TP + FN, consistency is 1. Consistency is calculated by the following equation.

$$Consistency=\frac{dn-k^2}{k(n-k)} \quad (12)$$

Here, d is the number of probable classes of error predicted by the model in each dataset (TP). K is the total number of classes prone to error in the data set (k = TP + FN) and N is the total number of samples.

11. Analysis of factor receptor indices

The analysis of factor receptor indices (FRIs) is an effective method for evaluating the performance of the prediction model. The characteristic curve of FRIs is defined as a plot of sensitivity on Y coordinate and, in the opposite, is one characteristic on the X coordinate. When FRI curve is under construction, many cutting points between 0 and 1 is selected, and the sensitivity and specificity at each cutting point is calculated. FRI curve is used to obtain the desired cutting point that maximizes both sensitivity and specificity.

FRI curve shows the advantages of using the model versus the cost of using the model in different threshold values. Indeed, FRI curve allows the evaluation of the performance of the predictive model in general and regardless of any particular cut-off value.

12. The level of area under the curve (AUC)

AUC can be deduced as a statistical descriptor to estimate whether the probability that a prediction model is to identify an error-bound sample is higher than an error-free sample. AUC less than 0.5 states that TP rate is very low. Therefore, AUC is used to evaluate the predictor's effectiveness.

13. Balance criterion

Balance is a criterion for measuring accuracy by calculating the Euclidean distance; the correct code is the best value, 1, and is often used by software engineers.

(13)

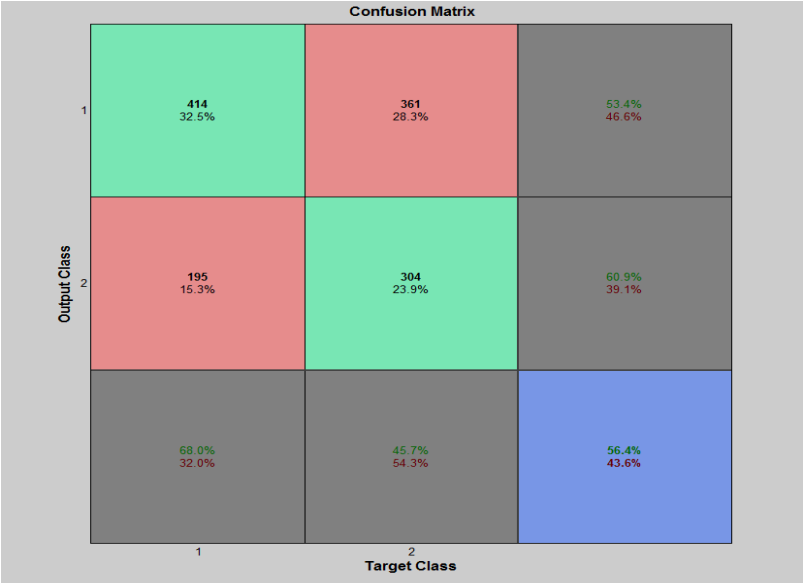
Artificial Neural Networks

The study of artificial neural networks is largely inspired by the natural learning systems where a complex set of neurons interconnected in learning work is involved. It is argued that the human brain is composed of 10^{11} neurons, where each neuron is associated with about 10^{4} other neurons. The speed of neuron transmissions is about 10^{-3} seconds, which is very insignificant compared to computers (10^{-10} seconds). However, one can detect a person's image in 0.1 seconds. This extraordinary power must be obtained from the parallel processing distributed in a large number of neurons.

An artificial neural network is a practical method to learn different functions like functions with real values, functions with discrete values and functions with vector values. A neuron alone can only be used to identify functions linearly separable. As in real problems, functions are not linearly separable, rather a network of neurons is needed.

Results

The Performance of Perceptron Neural Network (PNN)



$$B = 1 - \sqrt{\frac{1}{2}\left(\left(\frac{FN}{TP+FN}\right)^2 + \left(\frac{FP}{TN+FP}\right)^2\right)}$$

Figure 1: Confusion for PNN

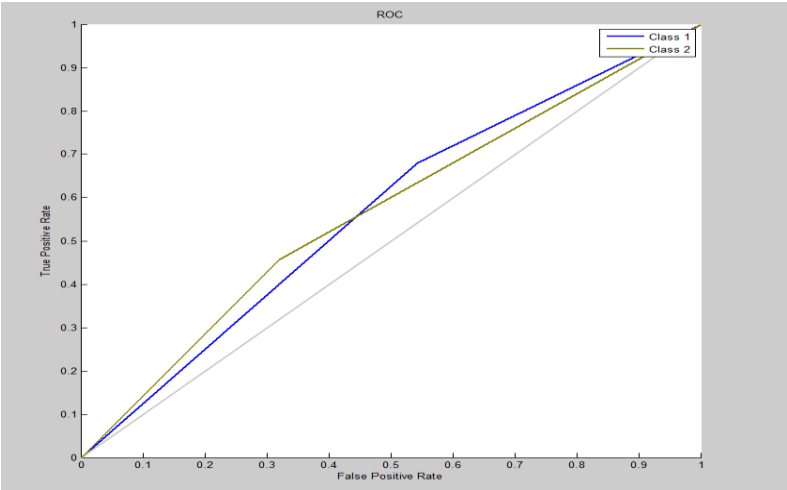


Figure 2: FRI for PNN

Table 1: Results obtained from combining PNN

MEAN		k1	k2	k3	k4
0.204693	False classification rate	0.25	0.1315789	0.210526	0.226667
0.28712	The cost of false classification	0.35913978	0.2168459	0.342509	0.229984
0.14356	Normalized cost of false classification	0.17956989	0.1084229	0.171254	0.114992
0.786342	Sensitivity	0.73333333	0.8888889	0.829268	0.693878
0.819709	Specific rate	0.77419355	0.8387097	0.742857	0.923077
0.709125	Precision	0.71	0.69	0.72	0.7165
0.862258	Accuracy	0.825	0.8888889	0.790698	0.944444
0.786342	Recalling	0.73333333	0.8888889	0.829268	0.693878

0.818721	F combined factor	0.77647059	0.8888889	0.809524	0.8
0.455014	Compatibility	0.34623656	0.7275986	0.629268	0.116954
0.803026	AUC	0.75376344	0.8637993	0.786063	0.808477
0.793244	Balance	0.75291736	0.8615077	0.781744	0.776809
4.080331	Run time	4.20269786	4.3184428	3.591441	4.208742

As the results show, the stability of PNN in all four runs have been indicated that the best result belongs to the third run. Thus, to this point the best accuracy has been obtained at 0.7091.

RBF system performance

For RBF system, MATLAB ready function was ued. In this function, the covered radius was 2. The results are as follows.

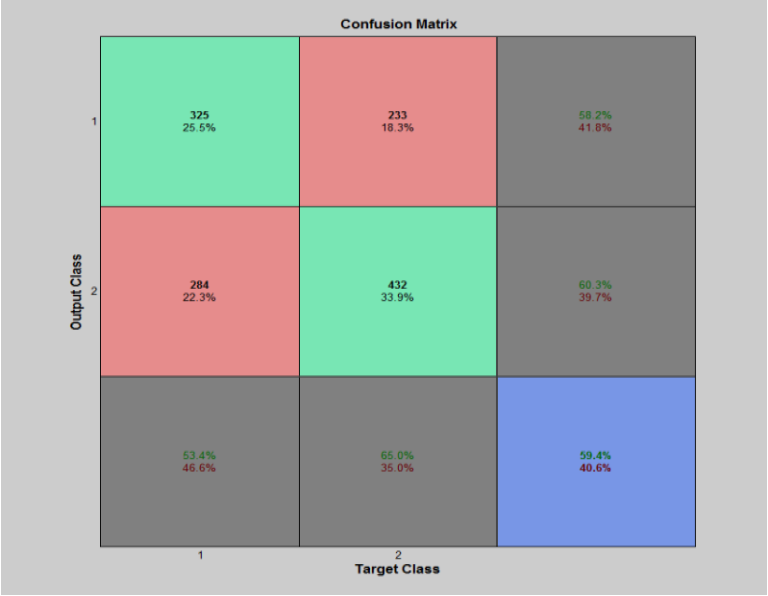


Figure 3: Confusion for RBF system

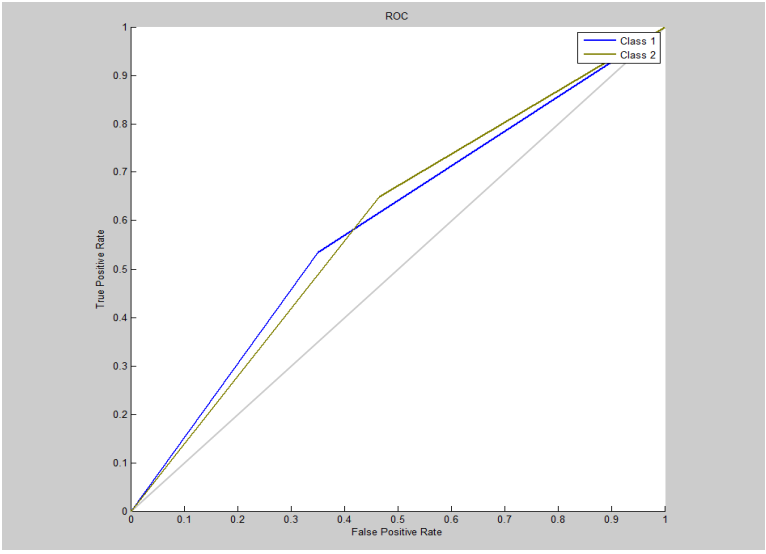


Figure 4: FRI for RBF system

Table 2: Results from the combination of RBF system

MEAN		k1	k2	k3	k4
0.432544	False classification rate	0.43421053	0.3947368	0.407895	0.493333
0.595086	The cost of false classification	0.62131148	0.6468531	0.458333	0.653846
0.297543	Normalized cost of false classification	0.31065574	0.3234266	0.229167	0.326923
0.562101	Sensitivity	0.55737705	0.6153846	0.583333	0.492308
0.623864	Specific rate	0.6	0.5454545	0.75	0.6
0.567456	Precision	0.56578947	0.6052632	0.592105	0.506667
0.90113	Accuracy	0.85	0.8888889	0.976744	0.888889
0.562101	Recalling	0.55737705	0.6153846	0.583333	0.492308
0.69116	F combined factor	0.67326733	0.7272727	0.730435	0.633663
-3.15608	Compatibility	-1.242623	-1.6573427	-6.91667	-2.80769
0.592982	AUC	0.57868852	0.5804196	0.666667	0.546154
0.589123	Balance	0.57814986	0.5789652	0.656408	0.542971
22.21198	Run time	27.8528068	23.630439	24.05727	13.30739

Thus, to this point, the best accuracy was at 0.605, which was a better result compared to the previous one.

Conclusion

The study dealt with the introduction and examination of two important prediction problems of content error and privacy, which are the predictive problems of error in the cloud-computing environment. One of the most significant problems in the online world is the problem of privacy, so that the concept of privacy is very different in various countries, cultures and jurisdictions. The privacy keyword includes the concept of data controller, data processor, and data subject. Overall, by observing the privacy problem, user trust increases and economic development prevails. The prediction problems such as the content error of data and preserving privacy were created in a cloud with certain cloud characteristics due to the openness and multi-tenant nature of cloud computing. The content-error data prediction and privacy protection in the cloud should be considered at all stages of the lifecycle of data. Given the results obtained, it is clear that the performance of RBF system is far better than the other two predictive systems.

References

1. Ghaffari. A.R. (2010). Obvious Computing Systems, Examples, Applications, Challenges. Master's seminar, Shahid Beheshti University.
2. Munir, K., & Palaniappan, S. (2013). Framework for secure cloud computing. Advanced International Journal on Cloud Computing: Services and Architecture (IJCCSA), 3(2).
3. Lee, J. H., Park, M. W., Eom, J. H., & Chung, T. M. (2011, February). Multi-level intrusion detection system and log management in cloud computing. In 13th International Conference on Advanced Communication Technology (ICACT2011) (pp. 552-555). IEEE.
4. Barford, P., Kline, J., Plonka, D., & Ron, A. (2002, November). A signal analysis of network traffic anomalies. In Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment (pp. 71-82). ACM.
5. Lakhina, A., Crovella, M., & Diot, C. (2004, August). Diagnosing network-wide traffic anomalies. In ACM SIGCOMM computer communication review (Vol. 34, No. 4, pp. 219-230). ACM.

6. Soule, A., Salamatian, K., & Taft, N. (2005, October). Combining filtering and statistical methods for anomaly detection. In Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement (pp. 31-31). USENIX Association.
7. Roschke, S., Cheng, F., & Meinel, C. (2009, December). Intrusion detection in the cloud. In 2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing (pp. 729-734). IEEE.
8. Mazzariello, C., Bifulco, R., & Canonico, R. (2010, August). Integrating a network ids into an open source cloud computing environment. In 2010 Sixth International Conference on Information Assurance and Security (pp. 265-270). IEEE.
9. Lo, C. C., Huang, C. C., & Ku, J. (2010, September). A cooperative intrusion detection system framework for cloud computing networks. In 2010 39th International Conference on Parallel Processing Workshops (pp. 280-284). IEEE.