Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 3, June 2021:138- 149

Research Article

Software Reliability Prediction Using Duane-CNN for Dynamic Webservices

Muzammil H Mohammed[a], Faiz Baothman[b]

Abstract:

Web services currently are based on integration of many services to provide response to user requests. When single service published there will not be more problem in aspects of utilizing effective response to user requests. We have analyzed there are certain complex arisen when we access recent days web service due to the nature of micro services for single applications and we ought to find the performance of quality parameters such as composability, security, reliability accountability and interoperability. To resolve above conflicts we need to measure mean time between failure(MTBF) or it is also known as reliability growth whenever changes are made in software design to make better web service. We proposed a scheme called D-CNN(Duane – Convolutional Neural Networks) to predict reliability growth using J.T.DUANE model to identify multiple failures during design and development of applications. The results of Duane model were given to CNN after proper training, pooling and other process to generate output. We have simulated above D-CNN and the results were compared with other existing models.

Keywords: *Web services, Reliability growth, multiple failures, prediction model and quality parameters*

[a,b] Department of Information Technology, College of Computers and Information Technology
Taif University, Taif, Saudi Arabia. m.muzammil@tu.edu.sa, f.baothman@tu.edu.sa
Received Accepted

Introduction

Reliability has become a basic issue with the always expanding size of present day applications, as these products are frequently exceptionally intricate and powerless against disappointment. To expand reliability, code defects in programming usage are known to be the essential drivers of disappointments. The expectation of programming abandons is a cycle of building classifiers to foresee code zones that conceivably contain deserts, utilizing information, for example, code unpredictability and history of progress. Forecast results will put cautions and agent their endeavors to code analysts. Records, alterations or strategies might be the code zones.[1][6]

The product's failure lead is represented by software reliability quality models. The models are used for quantitatively surveying the product. They evaluate the product's consistency by anticipating a product's imperfections or frustrations. Reliability quality is one among the product 's notable quality attributes during which the top user of software is more captivated than the product fashioner. The implementation of a programme can thus be enhanced by combining essential quality features like software reliability, maintainability and availability with execution properties like response time and throughput.[2][7]

As different users behave differently and therefore the communication links could also be unreliable, the length of every component device invoked may vary from one to a different. Additionally, when a neighborhood is named, reliability can fluctuate. so as to beat the above challenges, changes within the variable prediction duration should be handled by the web reliability prediction method. When a hard and fast prediction time is long enough, in most cases, we will break it into several intervals, referred to as the reliability statistic , and estimate the reliability of these time periods. In other words, for service-oriented SoS, the most objective of online reliability prediction is to forecast the statistic of the longer term time span, supported current system state and historical records.[3]

Current statistic model approaches exploit one user's time-dependent historical web service consumption data and use univariate statistic models.[8] The service quality, calculated from the customer side, are often influenced by the network or system latency of the user, which adds ambiguity to the dataset. Random uncertainty can affect the efficiency of the prediction process. Since an internet service is out there to the general public, the service are often visited by many users during an equivalent period of your time. The similarities within the use data of one web service from different users over an equivalent time could also be taken under consideration so as to attenuate the random error of the service quality data and to extend the prediction accuracy.[4]

Software reliability is characterized as the likelihood of a software system's failure-free operation for a specified duration in a specified environment. During operations, the failure of the programme may lead to client dissatisfaction, loss of market share, etc. The failure of a medical device or air traffic control system software can have a devastating impact on both the individual and society. It is also important that software companies ensure that their product is adequately reliable before releasing the software for use.[5][9][10]

Finally we need a proper mechanism to measure the various quality parameters which affects the performance of web services whenever the design constrains have been changed from existing service oriented architectures.web service is a kind of service returns the reliable information to the requested users and it is being upgraded whenever the scale up, design change, architecture style and other criteria to be measured. We strongly recommend checking the reliability growth of web service in terms of various QOS parameters. The main parameters we have considered here security, reliability, accountability, composability and interoperability.

Security is the key factor when published service information will be supplied to a requested user. User may be legitimate or unauthorized. When a service is accessed and utilized by an user means all the responsibility belongs to service provider described with the help of Web Service Description Language (WSDL).Composability and reliability are very important key parameters to monitor in web services in terms of access transparency an throughput time.

Interoperability deals with checking possibilities of a web server when it is being executed in different platforms like heterogeneous devices which are connected into the internet. Thus we need to check reliability growth whenever the design or any up gradation in existing web service. Figure 1 shows the reason behind the Duane model for multiple failures. We considered five parameters to check MTBF(Mean Time Between Failure) so single failure model is not suitable for our implementation of D-CNN proposed system.

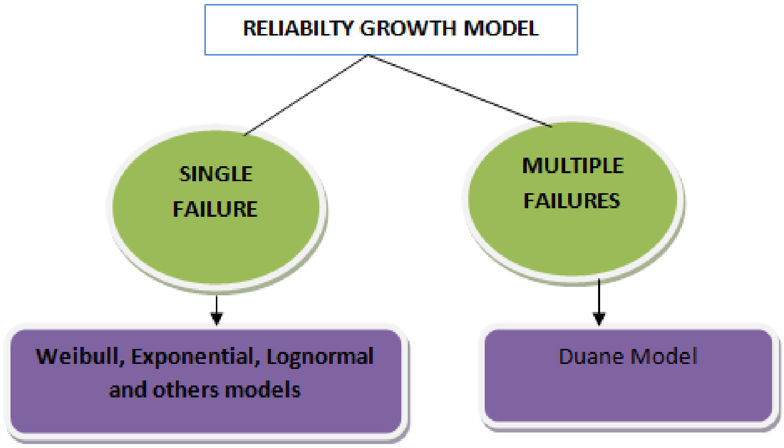

Figure 1: Reliability Growth Model

Related Works

Convolutional Neural Networks (CNNs) are specialised neural processing networks with a known grid-like topology, like 1D grid statistic data and 2D grid image data. In several practical fields, including speech recognition, image classification and natural language processing, CNNs are demonstrated to be effective. We leverage CNNs for efficient feature generation from ASCII text file during this work.CNNs' general architecture. CNNs have two main characteristics compared to standard Artificial Neural Networks (ANNs) or Multi-Layer Perceptrons (MLPs): Sparse Connectivity and Mutual Weights, which may enjoy our prediction of defects by capturing local programme structural information.[1]

Many AI models are utilized in software reliability standards with the event of software dependability research and AI. For device reliability prediction, an extended transient memory arrangement (LSTM) viewing system is usually recommended. The model beats the vanishing and detonating affectability of the straightforward recursive neural system for software reliability forecast, taking advantage of its particular information stream control structure. Additionally, the proposed solution joins layer standardization and truncates back proliferation. These two methods somewhat advance the impact of the model proposed. Numerical findings, compared with the simple recursive neural system, show that our proposed approach features a higher presentation and power as needed for software reliability.[2]

Nowadays, for reliability estimation, several neural network approaches are adopted. Few authors applied a wavelet neural network to the prediction of software reliability and proved the model's validity. For reliability prediction, they suggested a dynamic weighted combination model supported a rough feed forward neural network. One author proposed a replacement combination model of the neural network supported dynamic assessment weights, and proved that it's superior to the quality single model of the neural network. To predict the software reliability, few employed a back propagation qualified neural network (BPNN) and a community data handling system (GMDH). While these models demonstrate greater accuracy of prediction and adaptableness for reliable off-line data, the results of real-time data online reliability prediction are sometimes not ideal.[3]

Since statistic are like one another but not linear, during the standard statistic forecasting process, nonlinear patterns and nonlinear relationships should be treated. For this type of your time series, ARMA-related models aren't suited. There are some ways to construct non-stationary forecasting methods for statistic , like ARIMA and RNN.RNNs have advantages in handling the prediction of nonlinear and sophisticated patterns from historical data as a replacement emerging technology. a standard RNN, however, suffers from the vanishing gradient problem, and for the statistic forecast scenario, the long STM (LSTM) network is proposed. so as to construct statistic forecasting models, LSTM prediction techniques are borrowed.[4]

One of the ways the NHPP model is grouped is predicated on the shape of the mean function or the curve of fault detection. so as to predict software reliability, several researchers assumed that the fault detection curve would be an graph and developed exponential NHPP models. Later, researchers found that the programmer's awareness of the software package increases during the bug fixing phase of the life cycle of software development and as a result, the fault detection curve is presumably to be an s-shape when compare to other exponential model.[5]

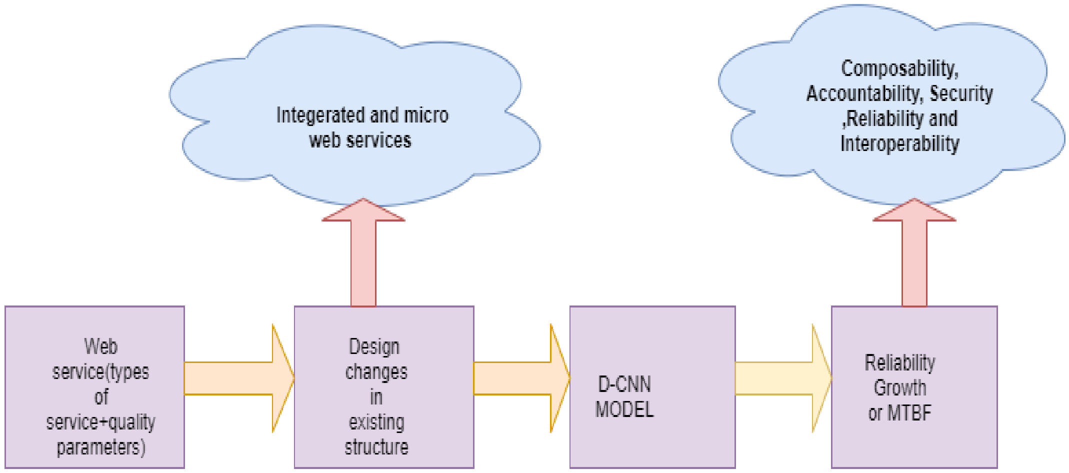

Figure 2: System representation of reliability prediction for web service

The above paragraphs have given enough knowledge about methods been used to identify software defect through various methods or models applied in web service up gradation and pre checking of the same used before deployment. Figure 2 shows the clear idea of proposed scheme in finding MTBF in web services. Recent days many micro services are being used to access a service and sometime these services can be integrated depends on the requirement of applications.

Before deployment of new service into the web we have to consider the nature of service and five quality parameters we discussed in introduction chapter. These parameters are applied to the Duane model to compare cumulative MTBF and instantaneous MTBF to find the necessity of reliability growth through Duane model implementation and the resultant values will be trained in hidden layers of convolution neural network(CNN) to predict the accuracy of the software is going to deploy in web services. The final predicted value will be represented in the output layer of D-CNN approach. [11]

Methodology and Implementation

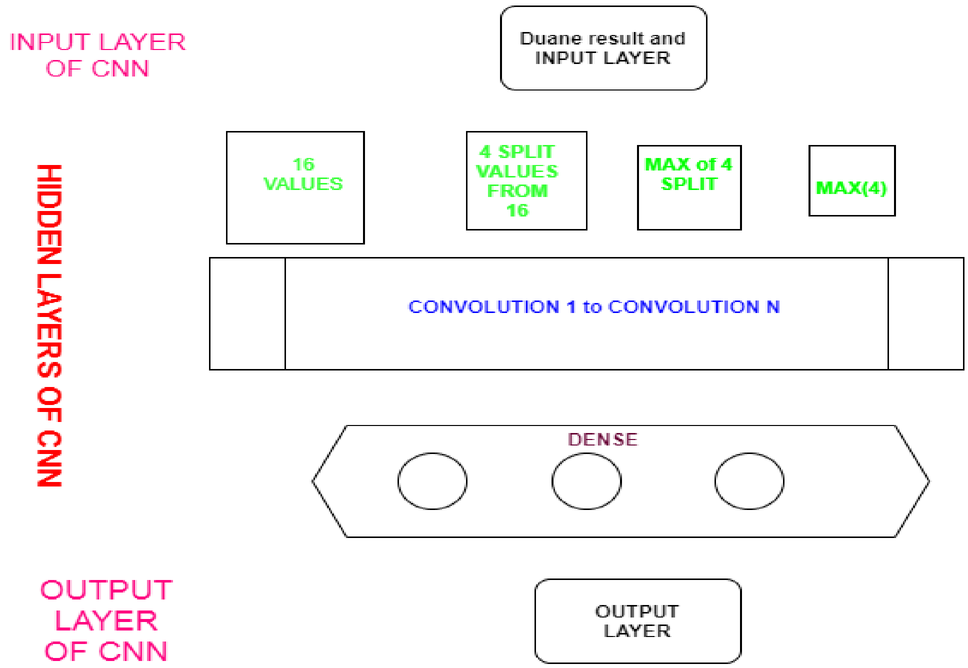

Figure 3: Overall system Representation for reliability growth prediction

Figure 3 shows the overall implementation of software reliability prediction using D-CNN for dynamic web services. Duane model prepared for 12 months and hours calculated based on everyday 6 hours working of new web service reliability prediction of failures of five QOS parameters. Cumulative hours and failures are calculated as per Duane specification and finally the result generated after comparing cumulative working hours (CWH) and CFQ(Cumulative failures quality parameters) obtained and log values are obtained. Finally cumulative MTBF of CWH found by subtracting slope value and graph generated by using above values and decided reliability growth increases in MTBF. The Table 1 shows the clear data transformation of final output prepared based on Duane model. Graph comparison also represented based on Table 1 and the same represented in Figure 4 and Figure 5.Figure 5 depicts clearly that we need to improve the MTBF for new web service considered in our simulation.

Table 1: Duane model for web service reliability prediction based of five parameters

Month	Cumulative Working hour(CWH)	Cumulative Failure quality parameters(CFQ)	Cumulative MTBF(CWH)	log(CWH)	log(CFQ)	Instantaneous MTBF(CWH)	Log(Instantaneous MTBF(CWH))
1	**180**	**1**	**180**	**2.255272505**	**2.255272505**	**179.9**	**2.255074615**
2	**360**	**2**	**180**	**2.556302501**	**2.255272505**	**179.9**	**2.255074615**
3	**540**	**3**	**180**	**2.73239376**	**2.255272505**	**179.9**	**2.255074615**
4	**720**	**4**	**180**	**2.857332496**	**2.255272505**	**179.9**	**2.255074615**
5	**900**	**4**	**225**	**2.954242509**	**2.352182518**	**224.9**	**2.352024213**
6	**1080**	**5**	**216**	**3.033423755**	**2.334453751**	**215.9**	**2.334288849**
7	**1260**	**6**	**210**	**3.100370545**	**2.322219295**	**209.9**	**2.32204968**
8	**1440**	**6**	**240**	**3.158362492**	**2.380211242**	**239.9**	**2.380062832**
9	**1620**	**8**	**203**	**3.209515015**	**2.306425028**	**202.4**	**2.306249129**
10	**1800**	**9**	**200**	**3.255272505**	**2.301029996**	**199.9**	**2.300851898**
11	**1980**	**9**	**220**	**3.29666519**	**2.342422681**	**219.9**	**2.342260777**
12	**2160**	**11**	**196**	**3.334453751**	**2.293061066**	**196.3**	**2.29287967**

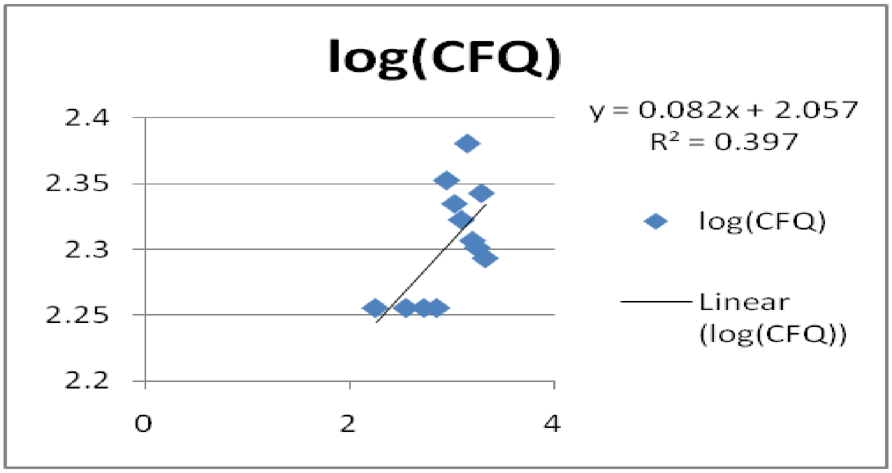

Figure 4: Cumulative failure QOS

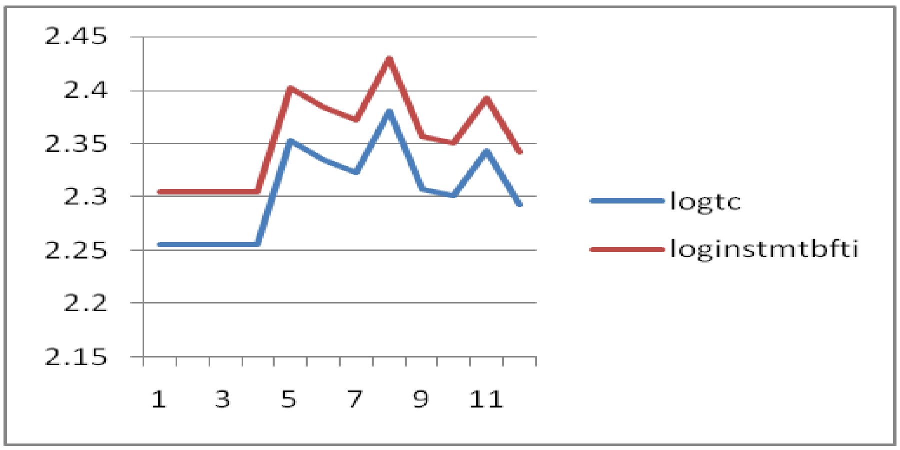

Figure 5: Instantaneous cumulative

We have implemented D-CNN reliability prediction model in TensorFlow Keras. As a sample code of convolution model illustration given below.

models = tf.keras.models.sequential{[….]}
model.compile(…..)
model.fit(……..)

The graph can be shown using **plt.show();** for comparing training and validation accuracy. The resultant values are supplied via of D-CNN input layer and data's will be given proper training through hidden layer of CNN process. It will predict clearly by manipulating again and again with the help of several convolution steps to find accuracy for example 0.3 percentage to 0.99 percentage of accuracy of finding accuracy of reliability growth. Hidden layer consists of pooling process like compression of data which represented in Figure 3.The matrix of 4*4 information perceived each 4 values and pick the highest from 4 split and find greatest value from each. Finally maximum values stored in output layers for accuracy. The Figure 6 has shown the accuracy of training and validation of our implementation of new web service reliability growth.

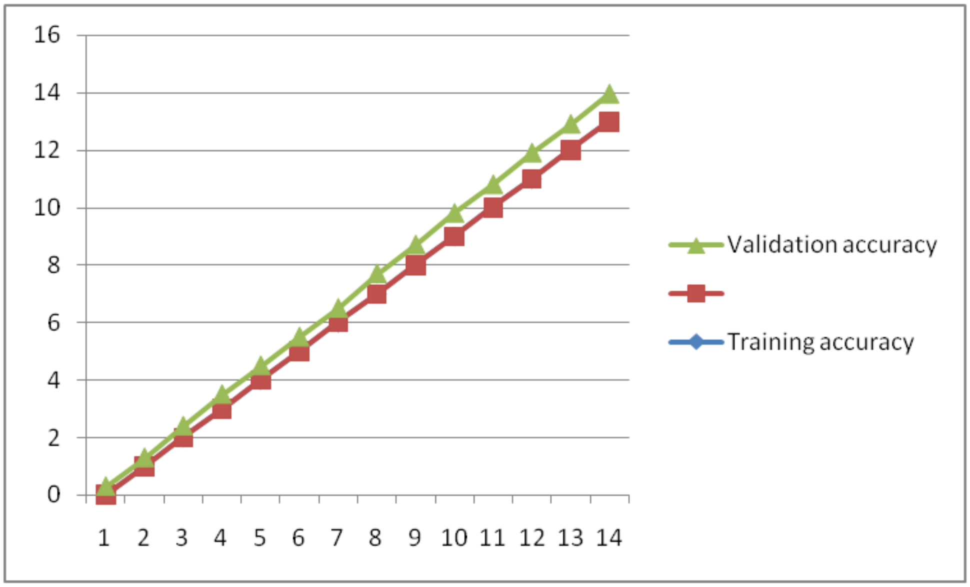

Figure 6: Validation and training accuracy

Conclusion

Web services are being deployed in web due to surge use of its benefits. So it needs to be upgraded very often because of drastic requirements of many business constraints. Mostly all kind of applications is migrated to online service and ease use of computing devices through internet connection. Several web services is to undergo the process of testing before published and subscribed by end user. These services contains many quality of service parameters as we discussed in earlier section and reliability growth is an essential before final version of complete new or expanded service. Thus reliability prediction of various attributes are mandatory and the same should be validated with the help of training data using various techniques like Duane and CNN model.

References

Li, J., He, P., Zhu, J., & Lyu, M. R. (2017). Software Defect Prediction via Convolutional Neural Network. 2017 IEEE International Conference on Software Quality, Reliability and Security (QRS). doi:10.1109/qrs.2017.42

Mayuri H.Molawade1, Dr. Shashank D. Joshi2, Software reliability prediction using Knowledge Engineering approach, International Journal of Advanced Trends in Computer Science and Engineering Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse14862019.pdf https://doi.org/10.30534/ijatcse/2019/14862019.

Hongbing Wang1a,Zhengping Yanga,Qi Yub,Tianjing Honga,Xin Lina , Online reliability time series prediction via convolutional neural network and long short term memory for service-oriented systems, Knowledge-Based Systems Volume 159, 1 November 2018, Pages 132-147, https://doi.org/10.1016/j.knosys.2018.07.006.

Ling Guo,1 Ping Wan,1 Rui Li,1 Gang Liu,2 and Pan He 2, Runtime Quality Prediction for Web Services via Multivariate Long Short-Term Memory, Hindawi Mathematical Problems in Engineering Volume 2019, Article ID 2153027, 14 pages https://doi.org/10.1155/2019/2153027

Boby John1, Rajeshwar S Kadadevaramath2, Immanuel A Edinbarough, A Brief Review of Software Reliability Prediction Models, International Journal for Research in Applied Science & Engineering Technology (IJRASET), Volume 5 Issue IV, April 2017 IC Value: 45.98 ISSN: 2321-9653

A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in International Conference on Neural Information Processing Systems, 2012, pp. 1097– 1105.

S. Honamore and S. K. Rath, "A web service reliability prediction using hmm and fuzzy logic models ," Procedia Computer Science, vol. 93, pp. 886–892, 2016.

J. D. Pfefferman and B. Cernuschi-Frias, "A nonparametric non stationary procedure for failure prediction," IEEE Transactions on Reliability, vol. 51, no. 4, pp. 434–442, 2002.

Zhang Xiaonan, YangJun feng, Du Siliang and Huang Shudong, A New Method on Software Reliability Prediction, Hindawi Publishing Corporation Mathematical Problems in Engineering Volume2013,ArticleID385372,8pages http://dx.doi.org/10.1155/2013/385372

Gusmanov, K. (2018). On the adoption of neural networks in modeling software reliability. Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering - ESEC/FSE 2018. doi:10.1145/3236024.3275433

JohnDonovana,EamonnMurphyb A new reliability growth model: its mathematical comparison to the Duane model, Microelectronics Reliability Volume 40, Issue 3, 17 March 2000, Pages 533-539,https://doi.org/10.1016/S0026-2714(99)00235-8