Journal of Computational Intelligence and Secure Systems of Artificial Intelligence
Volume 01, Issue 02, June 2025
www.jcissai.com

Design and Development of a Neural Network-Based Classifier Algorithm for Test Case Prioritization

M Suresh
Department of Electrical and Computer Science Engineering, Hitam Engineering College, Hyderabad
Corresponding Author: sureshcse@gmail.com

To Cite this Article
Suresh Naik, "Design and Development of a Neural Network-Based Classifier Algorithm for Test Case Prioritization", Journal of Computational Intelligence and Secure Systems of Artificial Intelligence, Vol. 01, Issue 02, June 2025, pp:10-12.

Abstract*: Test case prioritization has become widely accepted because it frequently produces high-quality software that is error-free. Traditional methods of prioritization are more expensive and time-consuming due to the rise in software errors. Because of the complexity of test suites and the lack of focus on automating the TCP process, the primary obstacle with TCP is the difficulty of manually validating the priorities of various test cases. This paper's goal is to identify the priorities of various test cases using an artificial neural network that uses the back propagation technique to forecast the right priorities. One such technique is used in our suggested work, where various test cases are given priorities according to how frequently they occur. Once the priorities have been allocated, the ANN predicts whether each test case will receive the right priority or not; if not, it generates an interrupt. Classifiers are used to categorize the various priority test scenarios. The suggested approach is highly successful since it automates the process of prioritizing the test cases and reduces complexity with strong efficiency.*

Keywords*: Classification, Software testing, Prioritizing test cases, Artificial neural networks, TF -IDF*

I. Introduction

The study and application of engineering to software design, development, and maintenance is known as software engineering. Software testing's primary goal is to assist designers and developers in creating a higher-quality, error-free system. Insufficient software testing results in software faults. To finish the testing process within time and financial restrictions, several tools, methods, and procedures are required for efficient and effective testing. Boost the possibility that the regression testing procedure may identify errors associated with particular code modifications.

II. Literature Survey

Prioritizing test cases is crucial for software testing in order to decrease regression testing time and increase the effectiveness of problem identification. A number of techniques have surfaced such as fuzzy logic, genetic algorithms and greedy algorithms. Because conventional approaches mainly depend on code coverage and previous faults, they sometimes fail to cope with changes in the software.

Research has mainly been on how neural networks can be used for smarter selection of test cases. To see how important test cases will be in the near future, neural networks, famous for their classification and recognition skills, study all previously made test executions and changes in the code. Neural networks have been shown to accurately order test cases when parameters like code complexity, past execution results and how likely the test cases find errors are considered.

Investigations have been carried out on using recurrent and feedforward neural networks for prioritization. They have demonstrated better accuracy and flexibility than the fixed rules commonly used. Having access to a wide range of high-quality data when making predictions is crucial, according to researchers. So, in terms of everything else, neural networks can be used to choose the right set of test cases, reduce testing expenses and ensure better software reliability in difficult and constantly shifting systems.

Although neural networks have been around for almost 50 years, they have just been used in the last 15 years, and the area is continually growing. Neural networks are made up of various components. The data contains some hidden, undiscovered information that neural networks can employ. Learning or training networks are used to capture

hidden information. By changing the values of the connections (weights) between pieces, we can train a neural network to carry out a certain task.

III. Current Areas

A document with a lot of words in it exists. Prioritizing words based on how frequently they occur in a document is necessary, and the back propagation algorithm is used to verify accuracy. Use classifiers to distinguish between words with the highest and lowest priority for simple prediction. There is currently no algorithm that uses neural networks for test case prioritizing, despite the implementation and discussion of several test case priority techniques. An artificial neural network aids in the proper assignment of priorities and produces an interrupt when various test cases are given incorrect priorities. Additionally, as no attempt has been done yet, the procedure needs to be automated.

Fig 1: Back Propagation Algorithm

Fig 2: Classification of Different Priorities

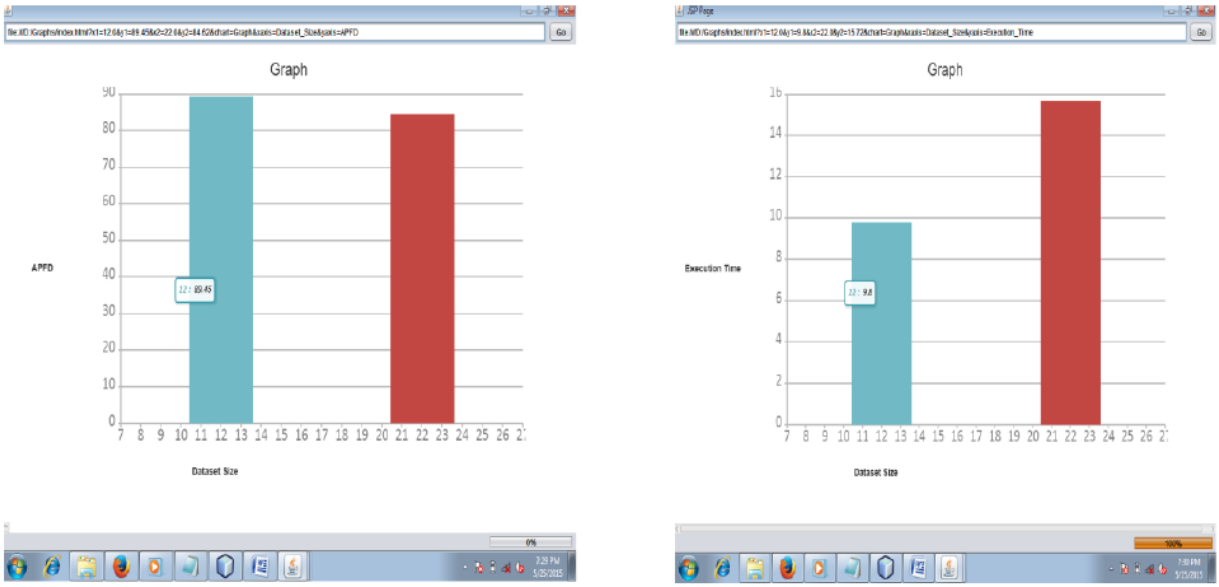

Fig 3: Execution Time

IV. Conclusion

This study presents a framework for prioritizing test cases based on TF-IDF, the naïve bayes classifier method, and the back propagation algorithm of artificial neural networks. The framework with embedded methodologies produced positive outcomes and validated our initial expectations and concepts. The algorithm is tested on a variety of publications. The framework was incredibly dependable and stable. Tests have shown that the implemented algorithm is sensitive. The number of unusable words in a document has a major impact on its classification, according to the content analysis, so improving document preparation is important to get better results. To quickly identify the highest and lowest priority words in a document, classification is crucial.

References

[1] Sokamso Tayang, Hubert and Deepak , Fuzzy Neural Networks: A Review from UPQC and Fuzzy Logic Statistical Perspective. Statistical science. 1998; 2(7): 12-18.

[2] Raju, Venkatesh, Hubert and Alwyn " Multilayer neural networks and PQ theory decision theory. *Science Direct*. 1991: 156-165.

[3] Guoqiang Peter Zhang. Neural Networks for Classification: A Survey. *IEEE Trans.On Man, Systems, and Cybernetics*. 2000; 30(4).

[4] G Rothermel, R Untch, C Chu, MJ Harrold. Prioritizing Test Cases for Regression Testing. *IEEE Trans. Software Eng*. 2001; 27(10): 929-948.

[5] P. C. Chang and C. H. Liu, "A TSK type fuzzy rule based system for stock price prediction," *Expert Syst. Appl*., vol.34, pp. 135–144, Jan 2008.

[6] SK Abdul Rehaman, John Vangli and Shanli, "Neural network system combined with Fuzzy-rough data reduction with ant colony optimization," *Fuzzy Set Syst*., vol. 231, pp. 56-65, March 2010.

[7] Chen Chen-Hung "A unctional-Link-Based Neurofuzzy Network for Nonlinear System Control"- *IEEE Transaction on Fuzzy Systems*, Vol 16 No 5, October 2008.

[8] Manju Bargavi, Siddharda Roy and Mohit Reddy, "ANN and FLANN Based Forecasting for Conceptual S&P 500 Index"*Information Technology Journal*, 6 (1): 121-132, 2010 Asian Network System for Artificial Scientific Information.