

# An Optimized Method of Module Selection During Software Development

Jasmeet Kaur, Gurpreet Singh

**Abstract:** The field of Software Engineering comes into the existence to avoid the damage caused by exploratory style of software development. Various steps specified during software development leads to the successful completion of the software under consideration. If the software is really not feasible then these steps will explain this in the initial investigation without wasting efforts in terms of time and person months. In this paper the focus is given on the Design and Implementation phases of the development process. A new method has been proposed to prepare a valid sequence of execution of modules to ease the work of implementation team. Convolution Neural Network has been used to find the best sequence of execution based on various coupling and cohesion parameters.

**Index Terms:** Design, Implementation, Neural Network, Software Engineering.

## I. INTRODUCTION

The era of technology leads to the development of untouched problems. Researchers are working hard to find the solution of all those problems, which required some solution to handle current issues. But now days, researchers almost achieved a recognizable target to address problems where manual work can be converted to increase the speed of the process and accuracy too. Now, this comment can be made that, new problem domain detection is very much difficult to target the set of tangible untouched problems. This directly means that now research direction is shifted towards the optimization of already suggested solutions. One of the game changer approach in the field of software development comes in to play i.e. Software Engineering in early 1960's [1]. Before the existence of this engineering approach, exploratory design approach was used for the purpose of development. During Exploratory design approach the coder has to think his/her own logics to handle the problem under consideration. This kind of approach may leads to the total disaster if any important point in the beginning of development is missed by the developer. This kind of failure will affect the total efforts in terms of person months and time wasted over development. The approach of software engineering provides a solution to avoid the disasters of exploratory style. The initial phases of the engineering approach i.e. Feasibility study of the problem under consideration and Requirement Analysis will help in this process. During feasibility study the problem under consideration is judged on the parameters like technical feasibility, Economic feasibility and User Acceptance.

**Revised Manuscript Received on July 02, 2019.**

Jasmeet kaur, Gurpreet Singh, Chandigarh University, Punjab, India.

During Technical feasibility, the focus is always on some logical issues which may require an extra efforts or focus may be given on hardware issues like the requirement of any

hardware device, which is not easily available. If the development team of the analyst after putting efforts will be able to provide the solution for these situations only then the software is supposed to be feasible, otherwise there must be a need to discard the idea of development to save extra efforts. Similarly, Economic feasibility just confirms the total cost required for the development of the software under consideration, if this cost increases the total benefits estimated after negotiation with the customer, again the development of the software is treated to be non-feasible. The last criteria during feasibility study i.e. User Acceptance plays a very important role. Because it directly deals with the end users. If initially the idea about the acceptance of end user has been grabbed then it will help in making the suitable working environment for the end users. If working environment is not suitable for the end users then again it leads to a non-valuable software and wastage of efforts. So, the above are some points which shows the importance of the software engineering approach over exploratory style. But again, in the proposed software engineering approach, when it comes to implementation contains lots of challenges. This paper focuses on one of these challenges i.e. what is the exact sequence of execution of modules in the Implementation phase. The next part of the paper will highlight the work performed during Design phase of the development cycle. This is followed by the challenges in Implementation phase of the development cycle. After that a dataset is explained which is used for finding the results by applying the current approach. Then there will be Results and Discussion section, which explains the effectiveness of the proposed method. At the end Conclusion will summarized the entire worked done behind the proposed approach and its implementation.

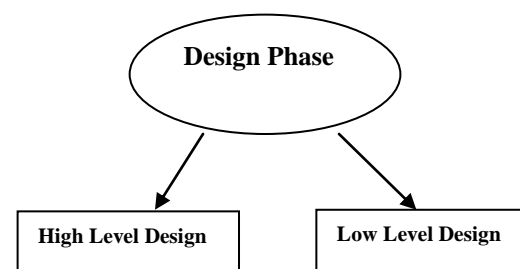


Fig 1: Design phase of software process

## II. DESIGN PHASE OF DEVELOPMENT CYCLE

During the Design phase of software development, the emphases is given to the identification of different isolated software modules and their working. This part covers two activities mainly as given below Fig1. shows the main activities of the Design phase too

### A. High Level Design

During this activity, the total number of modules in the



software has been recognized [2]–[4]. Then the relationship among modules is also identified. How modules are linked together i.e. which module's output is served as the Input to which another module etc.

### B. Low Level Design

This design activity involves in the preparation of the contract between various modules that how the modules can interact with one another. This mainly involves in the type of algorithms present in any module and what kind of data structure is used in isolated modules and during the exchange of information with other modules. It totally generates the idea about the type of coupling and cohesion used when two modules are in relationship with each other or how these are dependent on one another.

### III. THE CLASSIFIER: CONVOLUTION NEURAL NETWORK

Convolution neural network is a class of neural network. It contains input and output layers and multiple hidden layers. CNN carries multilayer perceptron's. Connectivity between neurons seems to be like visual cortex. Neurons respond to the stimuli only in receptive field. Every layer in multilayer perceptron is connected to another layer. A vector weight is chosen depending on the function applied to input values. Then these obtained values serve as the weights of Neural Network used and generate the implementation sequence of modules.

### IV. RELATED WORK

Pehna et al. [5] developed a framework known as iStar. They focused on the concept of strategic dependencies when try to consider modules for implementation. The next focus of their module was on strategic relationship between modules. Authors mainly worked on the interdependent parts of various modules of the software to deal with the strength with which modules are interacting with each other. Agent oriented approach was the key of concern in their research. Guerra et al. [6] proposed a new technique for testing for the implementation of randomized algorithms. Randomized algorithms are used in those kinds of applications where the input and the output parameters were non-deterministic or not easy to observe at each run of the algorithm. Brito et al. [7] showed the importance of Software process management (SPM) approach during the development process. They observed that the consolidation of SPM may strengthen the future development process. Singh et al. [8] proposed a framework for the identification of handwritten script. They proposed a module-based software interface for a handwriting recognition system and explain the coupling and cohesion strength of the proposed architecture.

### V. CHALLENGES DURING IMPLEMENTATION PHASE

The input to the implementation phase is considered as the information regarding the identified modules during the design phase. So, this can be considered as the number of modules to be implemented and the algorithms used during implementation. So, now the bigger challenge among the developer is to select the proper sequence of execution of the modules again this depends upon the type of problem under

consideration and the experience factor of the developers. Three categories of teams can be identified based on above two parameters as mentioned in Fig2.

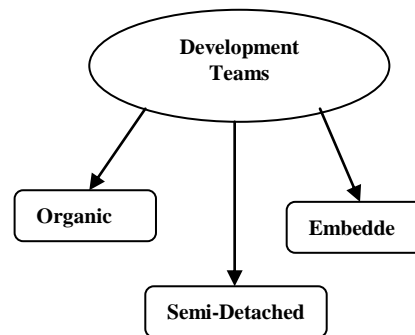


Fig 2: Types of Development Teams

Organic team composition always works on well understood problems and may have experienced or non-experienced developers in the team, Semi-Detached team composition is the situation when partial part of the problem under consideration is not clear to the developers but maximum problem definition is clear to them. This kind of team consists the composition of experienced and fresher's staff members. On the toughest end embedded team works. This team works on the problems which are complex in nature and not handled in the past. So experienced members for the development team are required [9].

To make this task an automatic one a new system is proposed as explained in Figure3. This system works on the concept of taking input from design phase as the number of modules and their relationship and then as output it will generate a sequence of execution of modules [10]–[12]. This sequence tells the developers from where to start and at which point, they have to end.

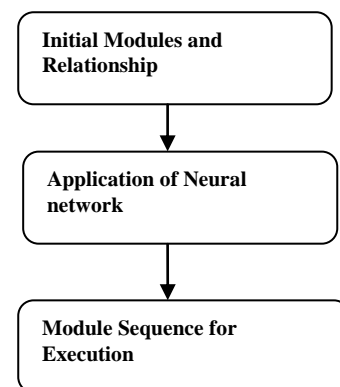


Fig 3: Proposed System

Neural network [13] is used by taking the initial sequence of modules as one of the initial population, then another random population can be taken and treated as another parent population. Then cyclic crossover operators can be used to make new children. After that new generated population is tested over a sequence of desired operations. Mutation is used to avoid the repetition of child generations.

## VI. RESULTS AND DISCUSSION

To find out the new sequence of execution of software modules NASA dataset of defense has been used [14]. This dataset is used for software defect prediction. Table 1. shows the results obtained in the form of Accuracy of execution sequence of modules the comparison parameters such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The overall accuracy obtained from the system has been considered as 97.70%. The different observations have been considered by increasing the number of modules at each run and by picking the random starting module each time. Graph 1. shows the accuracy levels at each observation and Graph2. shows the values of MAE and RMSE at each run of the application. The minimum values of coupling and maximum values of cohesion for different modules in combinations produce an accuracy parameter. This parameter further helps in chronological order to decide the exact sequence of implementation. This sequence never requires any skill dependency from implementation team and produces an exact order of implementation.

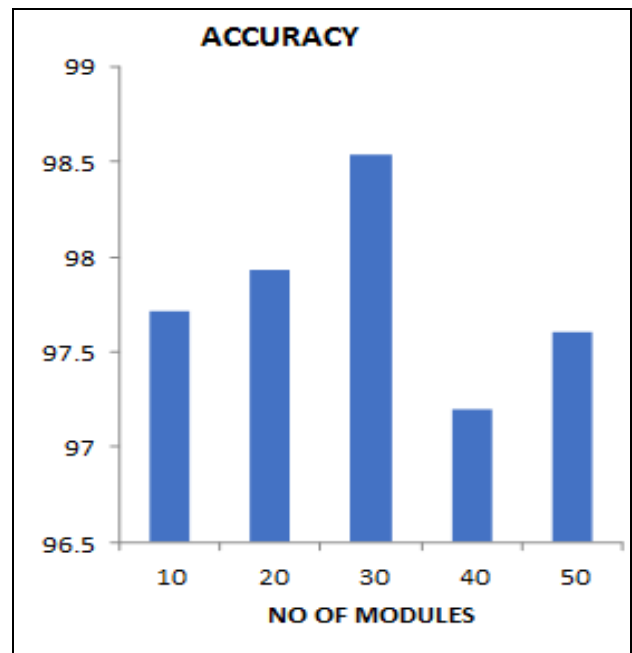
**Table I:** Performance on the basis of different modules by Neural Network

No. of Modules	Starting module	Accuracy	MAE	RMSE
10	50	97.71	12.57	15.97
20	60	97.93	12.45	16.66
30	70	98.54	12.65	16.2
40	50	97.2	11.82	14.93
50	50	97.6	13.75	16.73

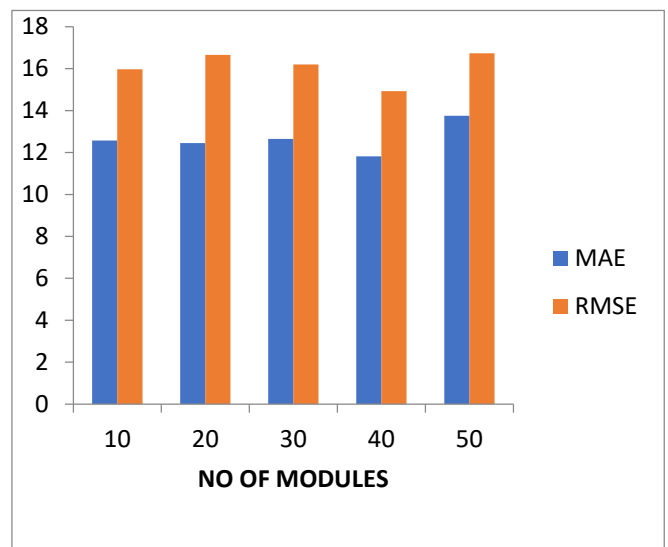
## VII. CONCLUSION

This paper proposed a new idea for the automation of sequence generation step during Implementation phase. With the help of information regarding the parameters of coupling and cohesion this new information can be generated [15]. This information is helpful in the cases as when the problems fall under the domain of semidetached and embedded categories of project. In these kinds of projects, expert opinion is must require. One cannot ignore the expert opinion as the problems are very complex in nature and not considered under the category of day to day problems. In this paper a solution is proposed, which is able to generate a new sequence of modules which diminish the requirement of expert opinion as the sequence generated by checking the coupling and cohesion parameters will leads to the successful development of the software under consideration. The overall accuracy of the developed system provides the exact sequence of

execution of modules, which satisfy the level of all types of feasibility checks.



**Graph 1:** Showing the performance according to accuracy.



**Graph 2:** Showing the performance according to MAE, RMSE.

## FUTURE SCOPE

In future more parameters can be considered other than coupling and cohesion to improve the overall performance of the system. The hybrid techniques or the combination of more than one classifiers always provide better results in comparison with unique or singly classifiers [16], [17]. So, in future the hybrid techniques can be implemented which may improve the accuracy level of the overall system.

## REFERENCES

- [1] Yurinda, *Software Engineering*, 2017.
- [2] G. Dougherty, *Pattern recognition and classification: an*



- introduction. Springer Science & Business Media, 2012.
- [3] A. Kumar and C. Ravikanth, "Personal authentication using finger knuckle surface," *IEEE Trans. Inf. Forensics Secur.*, vol. 4, no. 1, pp. 98–110, 2009.
  - [4] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," *J. Mach. Learn. Res.*, vol. 3, no. Mar., pp. 1157–1182, 2003.
  - [5] F. Penha, E. Miranda, M. Lucena, L. Lucena, F. Alencar, and C. S. Filho, "Actor's social complexity: a proposal for managing the iStar model," 2018.
  - [6] E. M. Guerra, S. M. Porto, J. Choma, and M. G. Quiles, "An approach for applying Test-Driven Development (TDD) in the development of randomized algorithms," 2018.
  - [7] D. F. Brito, M. P. Barcellos, and G. Santos, "Investigating measures for applying statistical process control in software organizations," *J. Softw. Eng. Res. Dev.*, vol. 6, 2018.
  - [8] G. Singh and M. Sachan, "A Framework of Online Handwritten Gurmukhi Script Recognition," vol. 8491, pp. 52–56, 2015.
  - [9] S. Ramteke, "Automatic Segmentation of Content and Noncontent based Handwritten Marathi Text Document," pp. 404–408, 2016.
  - [10] C. L. Huang and J. F. Dun, "A distributed PSO-SVM hybrid system with feature selection and parameter optimization," *Appl. Soft Comput.*, vol. 8, no. 4, pp. 1381–1391, 2008.
  - [11] D. Mateos-Garcia, J. Garcia-Gutierrez, and J. C. Riquelme-Santos, "On the evolutionary optimization of k-NN by label-dependent feature weighting," *Pattern Recognit. Lett.*, vol. 33, no. 16, pp. 2232–2238, 2012.
  - [12] A. Craig, O. Cloarec, E. Holmes, J. K. Nicholson, and J. C. Lindon, "Scaling and normalization effects in NMR spectroscopic metabonomic data sets," *Anal. Chem.*, vol. 78, no. 7, pp. 2262–2267, 2006.
  - [13] G. Singh and B. Singh, "Feature Based Method for Human Facial Emotion Detection using Optical Flow Based Analysis," *Int. J. Eng. Sci.*, vol. 4, no. 1, pp. 363–372, 2011.
  - [14] "NASA DATA SET for Software Defect Prediction." [Online]. Available: <http://promise.site.uottawa.ca/SERpository/datasets-page.html>. [Accessed: 07-Mar-2019].
  - [15] M. J. Lyons, "Automatic classification of single facial images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 12, pp. 1357–1362, 1999.
  - [16] G. Singh and M. K. Sachan, "Data capturing process for online Gurmukhi script recognition system," in *IEEE International Conference on Computational Intelligence and Computing Research (ICIC)*, 2015, pp. 518–521.
  - [17] G. Singh and M. Sachan, "Multi-layer perceptron (MLP) neural network technique for offline handwritten Gurmukhi character recognition," in *2014 IEEE International Conference on Computational Intelligence and Computing Research, IEEE ICCIC 2014*, 2015.

### AUTHORS PROFILE



**Jasmeet Kaur** received B.Tech. Degree in Computer Science & Engineering from Punjab Technical University, Jalandhar in 2012 and MTech Degree in Computer Science & Engineering from Punjab Technical University in 2015. She worked as Assistant Professor In Department Of Computer Science & Engineering KCCEIT Nawanshahr from 2016 to 2017. She worked as Assistant Professor in Department of Computer Science and Engineering from 2017 to 2018. Currently, she is working as Assistant Professor in Chandigarh University, Punjab. Her research areas include Digital Image Processing, and Software Engineering.

**Gurpreet Singh** received B.Tech. Degree in Computer Science & Engineering from

Punjab Technical University, Jalandhar in 2007 and MTech Degree in Computer Science & Engineering from BBSBEC in 2011. He worked as Lecturer in the Department of Computer Science & Engineering, IET, Bhaddal, Punjab from 2007 to 2010. He worked as Assistant Professor in the Department of Computer Science & Engineering, IET, Bhaddal from 2010 to 2014. He was a Research Scholar in CSE Department at SLIET, Longowal from 2014 to 2018. Currently, he is working as an Assistant Professor in Chandigarh University, Punjab. His research areas include Digital Image Processing, Handwriting recognition and Natural Language Processing. He is a lifetime member of Indian Society for Tech7.