

Minor Project Synopsis

On

Topic

“Software Fault Prediction”

Mentored By -

Mr. Sachin Garg & Mr Varun Goel

Student Details -

Student Name: Tushar Arora , Harshit Saini

Roll No: 20514803119 , 20214803119

Semester: 7th Semester

Group: 7I-8



Maharaja Agrasen Institute of Technology, PSP Area, Plot No. 1,

Sector – 22, Rohini, New Delhi – 110086

TITLE OF THE PROJECT

"Software Fault Prediction"

To develop or find algorithms with the help of machine learning to predict faults in software, which thereby increases software quality and testing efficiency.

PROBLEM STATEMENT

During recent years, we have witnessed a trend in which most of modern needs have become dependent on software. We can find it in our toothbrushes, watches, cars, etc. The IT and software industry has grown tremendously over the past few years, creating an increasing impact on the lives of people and on society as a whole. Consequently, we must make the software and applications more accurate, free of major errors, and more reliable. The more reliable the software is, the more valuable it becomes, and the less likely the software is to contain faults. Therefore, predicting software flaws could be very useful in the IT field and will have a profound impact on society at large.

Why is the particular topic chosen?

As stated in the problem statement, predictive fault detection is very important if we are going to improve software reliability and have a positive impact on the IT sector. Having been studying the IT field ourselves, we cannot miss the fact that we run into problems when using different software. It will be a great way to contribute to society as well as we will be learning a lot while developing and researching different solutions for the problem.

OBJECTIVE & SCOPE

The main objectives of this project are as follows:

- To determine algorithms using machine learning that can perform better in predicting the faults in software than the existing ones.
- To improve the results with the help of current datasets available on the internet using our algorithms.
- To use different evaluation benchmarks to evaluate the performance of our model.

The overall scope of the project is to create an automated software fault predicting model using machine learning techniques that will make the software more efficient and reliable to use.

METHODOLOGY

Software defect detection is mainly used for identifying the defective modules that are present in the software so that it can help in improving the quality of the software system. Machine learning techniques are useful in terms of software defect detection. Software is divided into defective and non-defective modules. For the classification of defective data sets, decision tree and logistic regression algorithms are implemented. The datasets are taken from the promised data repository and, thereby, accuracy is computed. The algorithms are implemented using the Weka tool as well as the Python IDE, and the comparative analyses of the results are displayed.

RESOURCES

Software used will be Jupyter notebook, Dataset using Kaggle, Python IDE, Machine Learning algorithms.

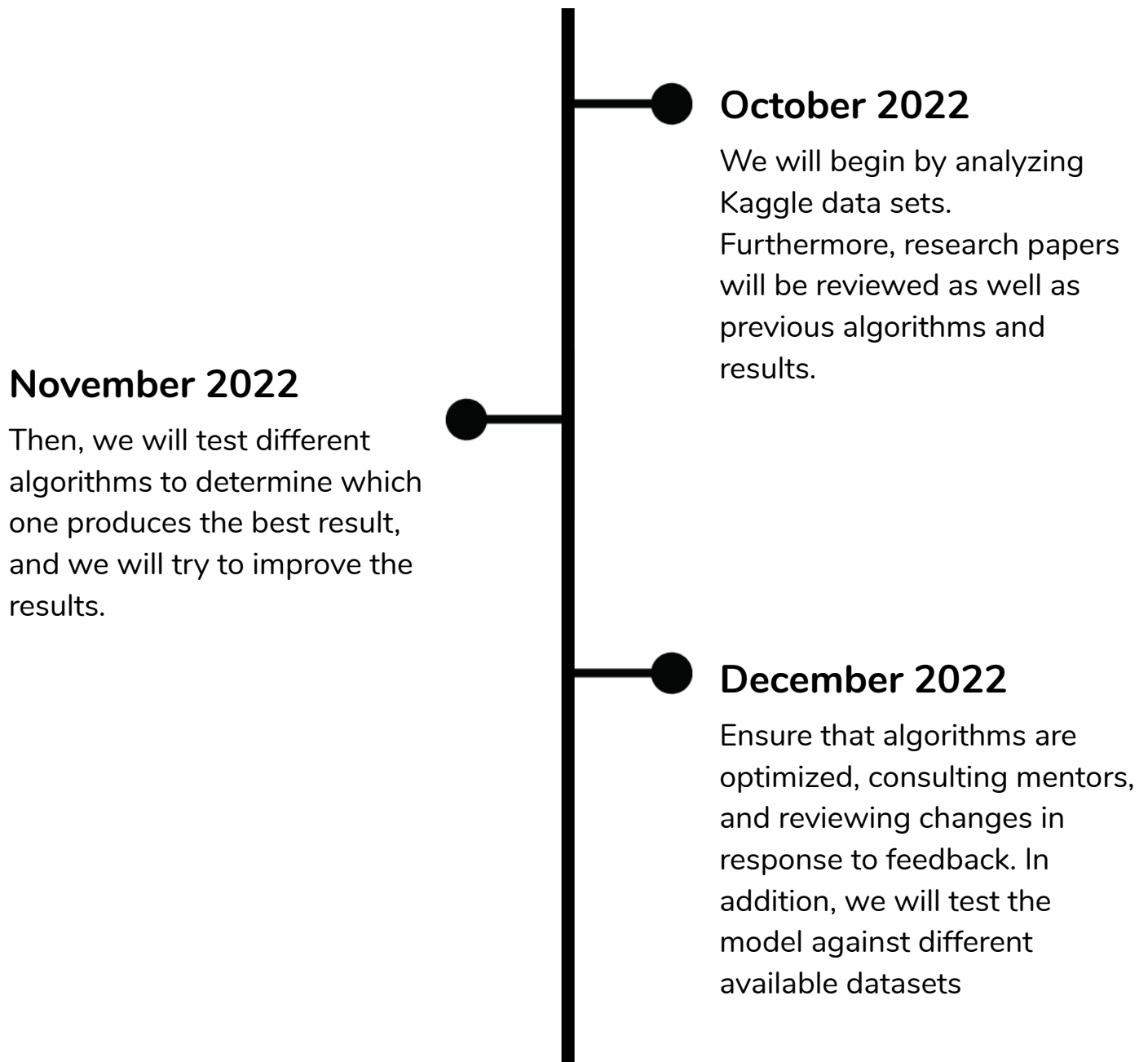
Hardware will not be used during this project.

CONCLUSION

What contribution would the project make towards the society?

We aggregated those parameters and analysed them accordingly, and we also illustrate the different challenges in the SFP domain. We also compared the performances of machine learning and statistical techniques based on SFP models. Our empirical study and analysis demonstrate that the prediction ability of machine learning techniques for classifying class/module as fault/non-fault prone is better than classical statistical models. The performance of machine learning-based SFP methods over fault susceptibility is better than conventional statistical purposes. The empirical evidence of our survey reports that the machine learning techniques have the capability to identify fault proneness and are able to form well-generalized results.

SCHEDULE OF THE PROJECT



REFERENCES & BIBLIOGRAPHY

- [1] C. Catal, U. Sevim, B. Diri "Software Fault Prediction of Unlabeled ProgramModules" Proceedings of the World Congress on Engineering 2009 Vol IWCE 2009, July 1 - 3, 2009, London, U.K.

- [2] Felix Salfner, Maren Lenk, and Mirosław Malek, "A Survey of Online Failure Prediction Methods," ACM Computing Surveys, Vol. 42, No. 3.
- [3] Yue Jiang, J. L., "Variance Analysis in Software Fault Prediction Models," IEEE 20th International Symposium on Software Reliability Engineering.
- [4] Du Zhang., Jeffrey J.P. Tsai., "Machine Learning and Software Engineering". Software Quality Journal, 11, 87–119.
- [5] M. Hecht, K. S. Tso, S. Hochhauser "The Enhanced Condition Table Methodology for Verification of Fault Tolerant and other Critical Software" .
- [6] Myron Hecht, Xuegao An, Bing Zhang, Yutao He "OFTT: A Fault Tolerance Middleware Toolkit for Process Monitoring and Control Windows NP applications" .
- [7] Golnoush Abaei, Ali Selamat "A survey on software fault detection based on different prediction approaches" - November 2013 .
- [8] Hassan Najadat, Izzat Alsmadi, "Enhance Rule Based Detection for Software Fault Prone Modules" January 2012 .
- [9] Yue Jiang, Bojan Cukic, Tim Menzies, Nick Bartlow "Comparing Design and Code Metrics for Software Quality Prediction" May 2008
- [10] T. Menzies, B. Caglayan, E. Kocaguneli, J. Krall, F. Peters, and B. Turhan, "The PROMISE Repository of empirical software engineering data," 2012. [Online]. Available: <http://promisedata.googlecode.com>.