Lov KUMAR, Anand TIRKEY, Santanu-Ku. RATH, 2018. An effective fault prediction model developed using an extreme learning machine with various kernel methods. *Frontiers of Information Technology & Electronic Engineering*, 19(7):864-888. https://doi.org/10.1631/FITEE.1601501

An effective fault prediction model developed using an extreme learning machine with various kernel methods

Key words: CK metrics; Cost analysis; Extreme learning machine; Feature selection techniques; Object-oriented software

Corresponding author: Lov KUMAR
E-mail: lovkumar505@gmail.com

Motivations

1. Software companies invest 50% of total software development costs in software quality assurance activities, in which defect-detection techniques are often applied to detect defective modules (Wanger, 2006).

2. There has been a lot of work in the area of fault proneness prediction using source code metrics and machine learning models. However, evaluating the usefulness of fault proneness prediction using cost factors is a less explored area.

Main ideas

1. Select a suitable set of source code metrics for fault prediction of object-oriented (OO) software.

2. Develop a fault prediction model using an extreme learning machine with various kernel methods.

3. Use a cost analysis framework to determine the usefulness of the fault-prediction model.

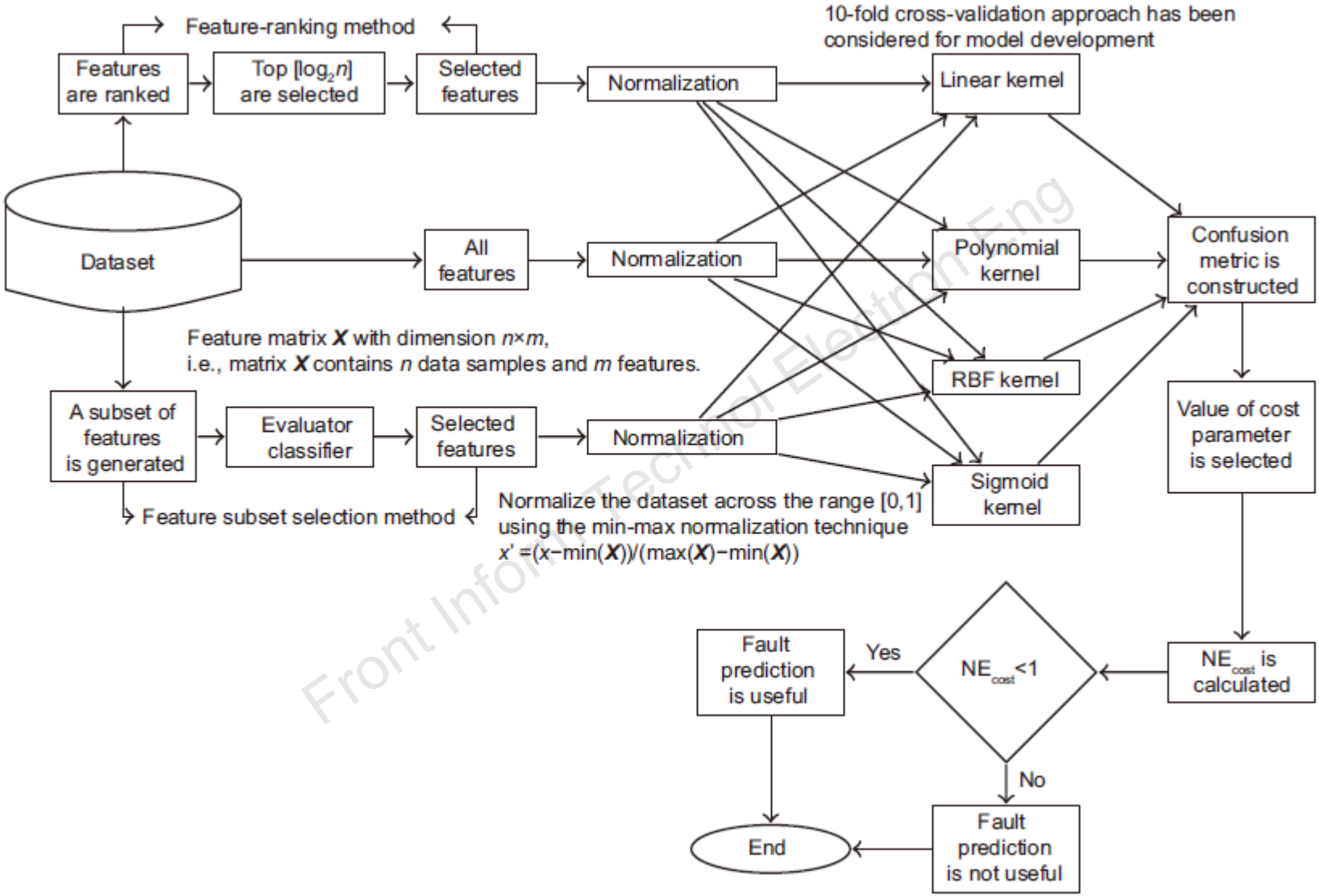


Fig. 2 Framework of the proposed cost-based evaluation framework (RBF: radial basis function)

Major results

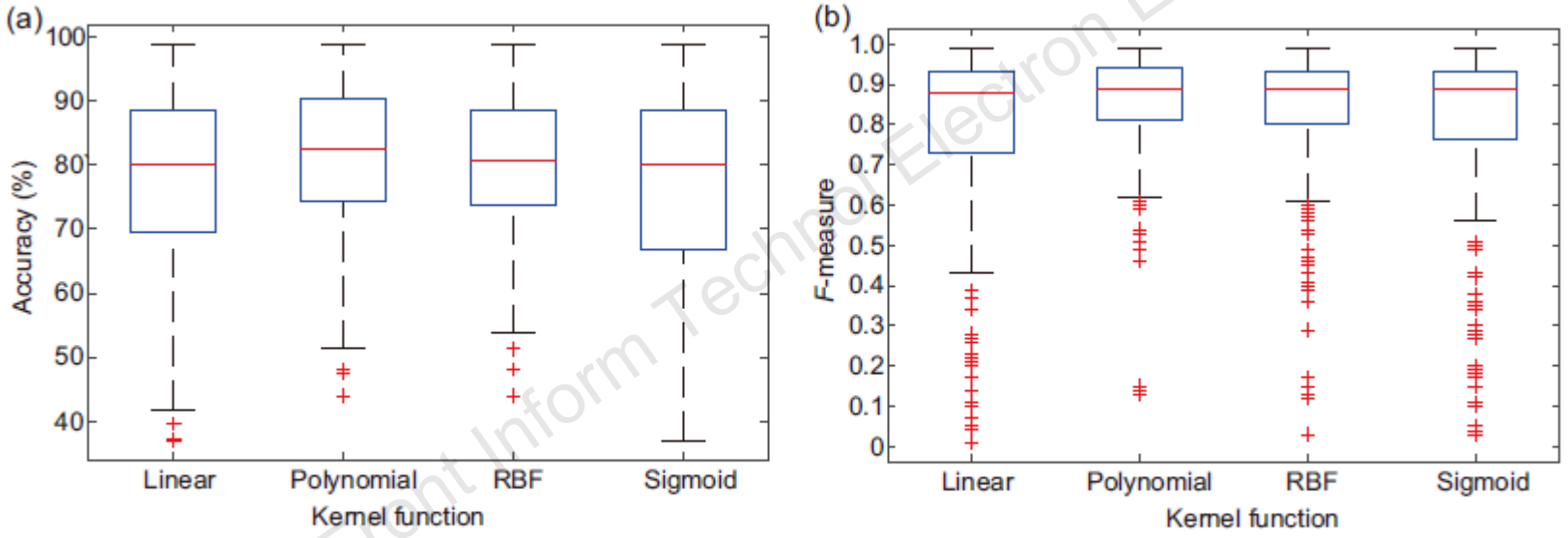


Fig. 7 Box plots displaying performance evaluation results for different kernel functions: (a) accuracy; (b) F-measure

Major results (Cont'd)

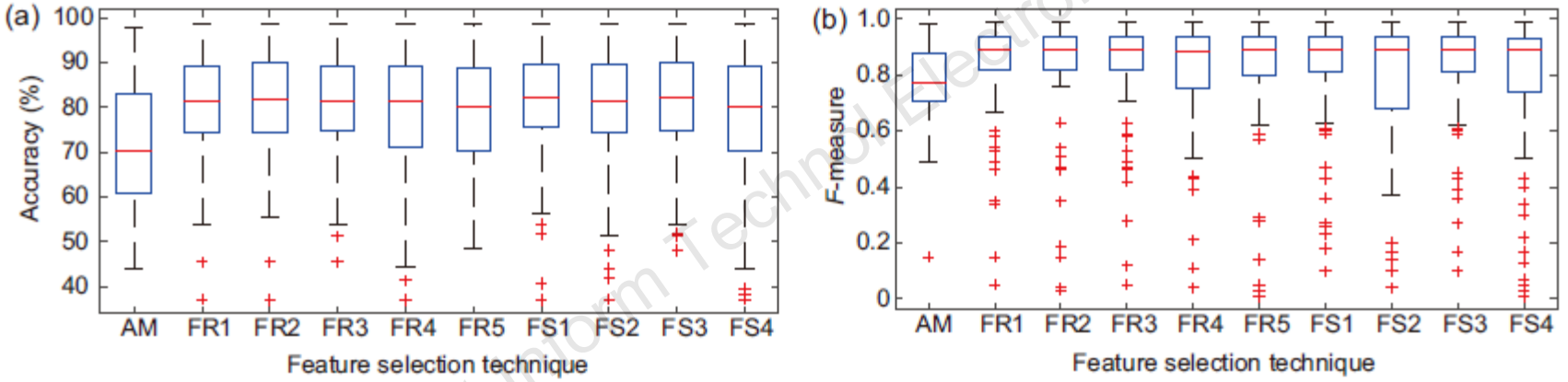
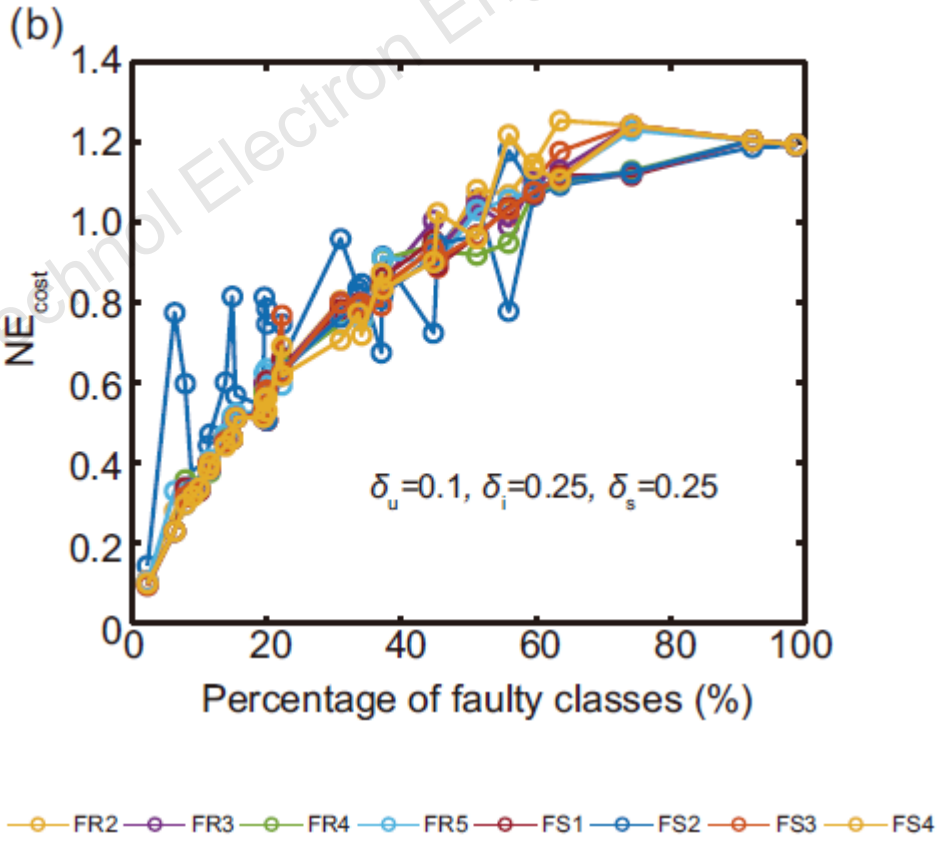


Fig. 8 Box plots displaying performance evaluation results for different sets of source code metrics: (a) accuracy; (b) F-measure

Major results (Cont'd)

It is observed that as the percentage of faulty classes increases, the fault-prediction technique tends to have a higher value of NE_{cost}; i.e., fault prediction can be useful for the projects with the percentage of faulty classes below a certain threshold.



Conclusions

1. Our experiment results suggest that it is possible to identify a small subset of OO source code metrics.

2. We found that the selection of classification method for developing a fault prediction model is affected by the feature selection methods.

3. Cost analysis results reveal that our developed fault prediction model is the best suitable for projects with the percentage of faulty classes below a threshold value depending on the fault identification efficiency (high: 25.72%; median: 39.24%; low, 47.28%).