DOI: 10.20396/revpibic.

Applying Convolutional Neural Networks to Adjust Code Complexity Metrics

Nykson S. Krenkin da Silva*, Plínio Roberto Souza Vilela.

Resumo

In the software development process, testing activities are responsible for identifying the presence of defects and allowing them to be repaired before delivery to the end user. Previous research on cost reduction strategies attributed to software testing, focused on test case prioritization techniques based on code size and complexity metrics in order to reduce the cost of software and still guarantee the ability to detect defects. However, these studies did not considered aspects related to the developer

Palavras-chave:

Software Testing, Software Defects, Complexity Metrics.

Introduction

The cost of software testing activity is significant in relation to the total cost of the development process.
The scarcity of resources, time and budget for development can lead to a serious problem which is the need to stop testing the software even before the minimum guarantee required for its specified quality.
The challenge is to establish a balance between revealing the greatest number of defects and do this with the lowest possible cost.
The innovative character of this proposal is the use of convolutionals neural networks to get information about the mood of a programmer while working on coding through the recognition and classification of their facial expressions. This information will be associated with the code developed and used to classify and estimate the number of defects in the software.

Discussion and Results

In this project we structured experiments to be carried out with programmers during development with the objective of capturing their biometric data and correlating them with the possible errors found in the code. The data used were the heart rate and facial expressions, which were later analyzed together with the recording of the screens of their computers, when they performed the development of the proposed problems.
In the process of analyzing the experiments we used a facial detection and classification API, where given an image of a face, returned the attributes of faces containing predictions based on neural networks and machine learning of facial features and the classification of the facial expressions detected in that Image.
We could observe that in situations where there was a need for some more complex logic to be developed in the code, there was some kind of alteration in the behavior of the developer, from the increase in the heart rate or some type of expression that showed apprehension and insecurity related to what was being coded at the time.
Precisely in those code snippets where there was a change in the behavior of the developers, the greatest number of errors occurred in the coding, which did not happen in other parts of the development where the behavior remained unchanged.

Conclusions

We believe that the developer's perspective on coding can be taken into account when it comes to software testing and reducing test costs. The project results have shown that the developer will somehow express some kind of discomfort when eventually encountering a laborious code snippet that may be defective and cause problems in the future.
We also consider that the application of the experiment on a larger scale, with a larger number of participants and using more data for the analysis, can more accurately assure the initial proposition of this research.

Thanks

Thanks to Prof. Plínio Vilela for all the guidance, commitment and patience for me and the project. And also thanks to the research group that contributed to the finalization of the project in the best possible way.

---

[1]MONARD, M. C.; BARANAUSKAS, J. A. Conceitos Sobre Aprendizado de Máquina. Sistemas Inteligentes Fundamentos e Aplicações. Barueri: Manole, 2003.

[2]LUCCA, L. W.; VILELA, P.; JINO, M. Avaliando a Relação entre Tamanho-Complexidade e Número de Defeitos de Software em Nível de Módulo. Lima: Jiisic, 2007.

[3]QIRONG, M.; DONG, M.; ZHAN, Y. Learning Salient Features for Speech Emotion Recognition Using Convolutional Neural Networks. Ontario: IEEE Transactions on Multimedia, 2014

[4] GAROUSI, V.; YASAMAN, A.; IRVING, R. A Search-Based Approach for Cost-Effective Software Test Automation Decision Support and an Industrial Case Study. Wageningen: IEEE International Conference on Software Testing, Verification, and Validation Workshops, 2014

[5] KORU, A. G.; ZHANG, D.; EMAM, E. K. An Investigation into the Functional Form of the Size-Defect Relationship for Software Modules. Baltimore: IEEE Transactions on Software, 2009

(cc) BY-NC-ND