Regression Testing of Database Applications Under an Incremental Software Development Setting

https://ieeexplore.ieee.org/document/8027014

Software regression testing verifies previous features on a software product when it is modifies or new features are added to it. Different approaches have been proposed to reduce the cost of regressuib testing, among which are: minimization, prioritization, and selection of test cases. Soft computing techiniques, such as data mining, machine learning, and others have been used to make regression testing more efficient and effective.

The research present a selection regression testing approach that utilizes a combination of unsupervised clustering with random values, unit tests, and the database schema to determine the test cases related to modifications or new features added to software products connected to databases. Their proposed approach is empirically evaluated with two database software applications in a production context. Effectiveness metrics, such as test suite reduction, fault detection capability, recall, precision, and the F-measure are examined.

Result of recent studies indicate that a more effective alternative for optimizing regression tests is to cluster the test cases according to some criterion, pattern or characteristic, such that test cases that detect the same fault are in the same cluster. [**Using Semi-Supervised Clustering to Improve Regression Test Selection Techniques**]

The method is to use an unsupervised clustering approach with radom values that selects a set of test cases. This research conduct an empirical evaluation of their approach with two datavase software products running in a productions setting.

Grouping test cases is used associated with data manipulation, particularly on the fields in the tables of a database. In successuve development increments, execute the group or groups of test cases that contain the test cases that have detected faults. The steps are the following: Test cases selection, Test cases similarity matrix construction, Test cases clusters generation, Clusters selection, and Clusters execution. Each step uses its own designed algorithm.

For some metrics, high values were obtained, however these values are balanced with others low values. The metric of detection capacity was 100%, and the precision was 10%. The values of the metrics for Silabo product are better than for the Estafeta product. This considering the sensitivity of the k parameter and the seed parameter in the clustering algorithm selected. The average metrics for each software product with regards to the effectiveness, the medium-sized software product (Silabo) has better results, with values of 40%, 100%, 12% and 20% for the reduction rate of the test cases, recall, precision and F-measure, respectively.

There are some threats to validity considered in the research. Internal threats can affect the validity of the results from the point of view of the execution of the empirical study. So the design and generation of the test cases after the development of the software product were considered. External threats: the regression testing approach on a small- and a medium- size software product, it is necessary to perform studies with large-scale software products. To reduce this threat, the test cases were generated by the same developers, and the faults considered were real. Threats to the construct are related to the metrics used in the empirical study. To reduce this threat, metrics related to the field of investigation were used.

In general, on average, the research proposed approach reduced the number of test cases to 14% in Estafeta product and 40% in Silabo product, both of them with a fault detection capacity of 100%, and on average with a precision of 10% and 12%, respectively.

I think the research was specific that they choose to focus on small- and medium- sized software products to test on. They used some methods and results from other researches for doing this research. To get these extended information helps to understand and design further studies.

Using Semi-Supervised Clustering to Improve Regression Test Selection Techniques
https://ieeexplore.ieee.org/document/5770589?arnumber=5770589

Cluster test selection is proposed as an efficient regression testing approach. It uses some distance measures and clustering algorithms to group tests into some clusters. Tests in a same cluster are considered to have similar behaviors. All existing cluster test selection methods employ unsupervised clustering. The previous test results are not used in the process of clustering. It may lead to unsatisfactory clustering results in some cases.

In this research, a semi-supervised clustering method, it is to improve cluster test selection. It uses limited supervision in the form of pairwise constraints: Must-link and Cannot-link. They are derived from previous test results to improve clustering results as well as test selection results. Test selection using clustering mainly contains four steps: Capturing feature, Distance measure, Cluster analysis, Sampling strategy.

The results are semi-supervised K-means can improve the test selection results in most versions for both programs, Flex, and Space. Although is has different effectiveness on different individuals, the comprehensive results are consistent. The rate of failed tests will affect cluster tests selection with semi-supervised K-means. It has a better effectiveness when the failed tests are in a medium proportion. The constraint definition will also affect cluster test selection with it. A strict definition can usually achieve a better effectiveness in most cases.

According to the authors, threats to external validity include the use of only a small set of subject programs, modified versions, and test sets. Threats to internal validity include the correctness of collecting execution profiles, clustering processes and constraint derivation etc.

The conclusion is cluster test selection with semi-supervised K-means has a better effectiveness when the failed tests are in a medium proportion, and a strict definition of pairwise constraint can improve the effectiveness of cluster test selection with semi-supervised K-means.The experiment results of this research indicate that the semi-supervised clustering method sumi-supervised K-Means can improve test selection in most cases.

Because this is the first research using semi-supervised clustering for regression test selection. In the future work:

1. Semi-supervised K-means is good at preserving the intrinsic structure of data as well as dimensionality reduction. Hence semi-supervise K-means is not suitable for the application with a small number of features. There are many semi-supervised clustering methods and they would be studies for test selection in further.
2. The clustering results depend on high quality constraints. So it is interesting to build high quality constrains for different testing scenarios.

I think this related research gives a wide range of opportunity of improving regression test selection in other ways.