Towards Machine Learning Based Design Pattern Recognition

Paper link: https://ieeexplore.ieee.org/abstract/document/6651312

1.Summary

1.1 Motivation

This approach aims to optimize the performance of machine learning, especially artificial neural networks (ANNs), for design pattern recognition (DP). The goal is to learn recognition rules directly from concrete DP instances to address issues such as rule granularity and abstract DP attributes. This method avoids default rules and uses feature selection to improve accuracy. To overcome the lack of reference standards, a diverse training dataset was created based on 400 open-source applications analyzed with DP-aware tools. This innovative approach addresses the problem of underutilization of ANNs in DP recognition and improves accuracy by learning from real-world examples instead of relying only on theoretical explanations.

1.2 Contribution

We discuss different approaches to design pattern recognition (DP) using artificial intelligence (AI) technology. One method uses graph theory to identify models with submodels and to represent uncertainty using fuzzy beliefs. Another uses statistics based on theoretical DP properties and subsequently refines them through machine learning to improve accuracy. An alternative approach is to build a DP model on the fine structure using graph matching and classifiers to filter out false positives. Other methods use training models to reduce the search space. The discussed methods differ in that they use graph-based techniques for AI and DP recognition.

1.3 Methodology

This methodology addresses the design pattern (DP) recognition gap by building a comprehensive feature set that exceeds theoretical specifications, consisting of 82 indicators in the first stage and 112 indicators in the second stage. Object-oriented statistics are calculated using CKJM2, JMT3, POM4 and Dependency Finder5. Feature selection methods such as ReliefF, Fisher, Gini index and spectral identify optimal subsets for different DP roles. The unbalanced dataset problem is solved by using ADASYN for synthetic sampling. An artificial neural network (ANN) with backpropagation training is used and several configurations are tested against the best topology model determined by 10-fold cross-validation. We outline a systematic approach to feature selection, dataset processing, and model training in DP recognition.

1.4 Conclusion

The paper presents a new design pattern (DP) recognition method using artificial neural network (ANN) and feature selection techniques. Unlike traditional methods, our method derives recognition rules and features from real cases rather than theoretical explanations. Experimental results recognizing six DPs in open source applications show promising and comparable results, confirming the feasibility of ANNs for solving DP recognition software problems.

2. Limitations

1.This work faces significant limitations due to the lack of standardized benchmarks to objectively evaluate design pattern recognition (DP) approaches. Manually checking results without such benchmarks is common in the literature, introducing subjectivity. In response, this paper attempts an objective assessment using the PMARt repository, a peer-reviewed resource. However, there are concerns about the potential imperfections of the PMART that affect the completeness of the assessment.

2.Another important limitation acknowledged in this article is the inherent subjectivity when evaluating PD recognition approaches. Relying on manual inspection introduces variability as different assessors may label the same sample differently. To facilitate this, the article takes an objective approach rather than relying on the author's judgment. Instead, we rely on the PMARt repository, complete with the internal documentation of the analyzed application (JHotDraw 5.1) and related literature. However, reliance on external sources creates dependency and potential limitations on the completeness and accuracy of the references selected.

3. Synthesis

Plans include exploring alternative learning methods for artificial intelligence, such as support vector machines (SVMs) and network-based adaptive fuzzy inference systems (ANFIS). We use learning and rule extraction methods and compare the extracted rules with those derived from theoretical explanations.