Cornering Cohesion: Investigating new ways to measure software cohesion

Master's Thesis

Supervisor: Prof. Dr. Alexander Pretschner
Advisor: Markus Schnappinger, Dr. Arnaud Fietzke (itestra)
Email: {pretschn, markus.schnappinger}@in.tum.de
Phone: +49 (89) 289 - 17386
Starting date: immediately

Keywords

Machine Learning, Data Mining, Natural Language Processing, Software Quality

Context

Software vendors aim to develop software systems that fulfill all functional requirements, are cheap to build in the first place while also being easy to maintain in the future. Over the past decades, several programming and design paradigms have emerged to ensure *good design* and *easy maintenance*. Most popular among these are e.g. the concepts of low coupling and high cohesion. A multitude of metrics like LCOM, LCOM2, or LCOM5 were developed to measure high and low cohesion [1,2,3], though there is no consensus which metric captures the notion of cohesion best [4]. Hence, the usefulness of these metrics in the context of quality analysis is unclear [5].

Goal

The objective of this thesis is to develop new ways to measure functional cohesion and examine the correlation of these metrics with the actual maintainability of a program. This analysis provides valuable insights which metrics should be used in automated maintainability analyses and which are not recommendable.
For example, Latent Dirichlet Allocation (LDA) is a technique to extract topics from documents written in natural language and group them according to these topics [4]. Applying an adapted version of this algorithm, the student develops a method to group programs and subprograms according to topics appearing in source code. This method can then be used to design a new cohesion metric. In addition to LDA, other Natural Language Processing (NLP) techniques can be taken into account. For example, the similarity of identifiers or their wordnet distance are promising features.
In a second step, the student incorporates information from the NLP techniques with other available information, e.g. the callgraph of the program. For example, callgraphs provide insights where the output of a method is actually used. Also, graph network metrics are likely to obtain useful information. Finally, the student investigates the relationship between the novel algorithms, existing measurements and the perceived cohesion.

Working Plan

1. Familiarize yourself with the concept of cohesion and cohesion metrics
2. Applying clustering algorithms for source code based on topics (LDA)
3. Develop clustering algorithms for source code based on other NLP features
4. Extract graph representations of programs and compute relevant graph metrics
5. Integrate graph features and NLP features
6. Propose new cohesion metrics
7. Compare the usefulness of the developed metrics and existing metrics for quality analyses

Deliverables

- Source code of the implementation.
- Technical report with comprehensive documentation of the implementation, i.e. design decision, architecture description, API description and usage instructions.
- Final thesis report in English written in conformance with TUM guidelines.

References

[1] S. R. Chidamber and C. F. Kemerer, Towards a metrics suite for object oriented design. 1991
[2] Emerson, Thomas J. A discriminant metric for module cohesion. 1984.

Fakultät für Informatik
Lehrstuhl 4
Software und Systems Engineering
Prof. Dr. Alexander Pretschner

Boltzmannstraße 3
85748 Garching bei München

Tel: +49 (89) 289 - 17386
https://www4.in.tum.de

sponsored by: itestra be excellent

Application:
Please apply via email to markus.schnappinger@in.tum.de. Your email should explain your interest in the topic and contain your current transcript of records. Upon mutual agreement, the thesis can be sponsored by itestra GmbH.

[3] Fernandez, Luis and Pena, Rosalía. A sensitive metric of class cohesion. 2006.
[4] Etzkorn, Letha H., et al. "A comparison of cohesion metrics for object-oriented systems." Information and Software Technology 46.10 (2004): 677-687.
[5] Briand et al. A unified framework for coupling measurement in object-oriented systems. 1999
[6] Briand et al., Exploring the relationships between design measures and software quality in object-oriented systems. 2000

Fakultät für Informatik
Lehrstuhl 4
Software und Systems Engineering
Prof. Dr. Alexander Pretschner

Boltzmannstraße 3
85748 Garching bei München

Tel: +49 (89) 289 - 17386
https://www4.in.tum.de

sponsored by:

Application:
Please apply via email to markus.schnappinger@in.tum.de. Your email should explain your interest in the topic and contain your current transcript of records. Upon mutual agreement, the thesis can be sponsored by itestra GmbH.