Assessing the Quality of Code Comments

Master's Thesis

Supervisor: Prof. Dr. Alexander Pretschner
Advisor: Markus Schnappinger, external advisor from itestra
Email: markus.schnappinger@tum.de
Phone: +49 (89) 289 - 17386

Starting date: immediately

Keywords: Data model, Design, Software Quality

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

Context

The underlying data model is the basis for the whole information system. Any design decision made here strongly influences further development and system performance as a whole. To foster good design of such models, Matthew West published a set of principles that should be respected during the design of a data model [7]. David Hay also provides guidelines in his book "Data Model Patterns" [1]. Such design principles are good guidelines for conceptualizing new data models based on given requirements. When working with proposed or existing data models, however, it can be hard to accurately grasp and communicate the quality of a data model.
There is a need for objective evaluation of the quality of such models. Hence, Moody proposed an evaluation framework consisting of quality attributes, metrics to measure these attributes, a weighting function, and strategies to overcome detected quality flaws [3]. However, in later research Moody found many quantitative measures to be inferior to subjective ratings of individuals, stressing the need for practically usable metrics [2]. Current software quality assessments [4, 5] respect the data model as well, but have little tool support. Though static analysis can automatically extract data types, count attributes or tables, further research is needed to (i) define a quality model for data models in the first place and (ii) support the evaluation of such models.
With suitable metrics in place, strategies can be developed to improve on deficiencies in one or more quality dimensions. Eventually, metrics should be able to facilitate the development of concise, consistent and flexible data models as the basis of system design.

Goal

This thesis aims to investigate methods for automated quality analysis of data models. In a first step, important quality dimensions and their impact on system performance are explored from literature. Building on that, different metrics for good quality are examined and evaluated w.r.t their usefulness on real world examples. The metrics should be objective and be compiled automatically. This means they should express characteristics inherent to the data model, which can be determined even without detailed knowledge of the requirements for a specific use case. Alternatively, user interaction can be used to create a context aware metric. In addition to traditional static analysis, novel methodologies like statistical analysis, semantic identification and machine learning techniques can be explored. Possible fields of interest entail:

- ranking and analysis of relationship graphs,
- semantic disambiguation of model entities based on a domain ontology [6],
- data analysis for functional dependency estimation and
- process analysis (e.g. heat maps of common data access patterns).

Eventually, the performance of the approaches is evaluated with either open source projects or example code provided by the industry partner.
This research is going to be conducted in cooperation with itestra GmbH. If you are interested in this topic, please follow the application instructions below.

Working Plan

1. Familiarize yourself with software quality prediction using ML
2. Research which text-based attributes of code can be used as ML features
3. Use these features to predict code readability and understandability in a prototypical implementation
4. Evaluate which algorithms perform best to predict these quality attributes
5. Investigate the reasons for performance differences and identify potential for improvement

Application:
Please apply via email to markus.schnappinger@tum.de. Your email should explain your interest in the topic and contain your current transcript of records. The most promising candidates will be invited for an informal interview. Upon mutual agreement, the thesis can be sponsored by itestra GmbH.

6. Create screenshots of texts and apply image-based machine learning algorithms to predict readability and complexity
7. Compare the performance of image-based and text-based approaches
8. Record the studies, results, findings and the engineering approach in form of a thesis

Deliverables

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

References

[1] David C. Hay. Data Model Patterns: Conventions of thought. Dorset House, 1996.

[2] Daniel L Moody. Measuring the quality of data models: an empirical evaluation of the use of quality metrics in practice. ECIS 2003 Proceedings, page 78, 2003.

[3] Daniel L Moody and Graeme G Shanks. What makes a good data model? evaluating the quality of entity relationship models. In International Conference on Conceptual Modeling, pages 94–111. Springer, 1994.

[4] Markus Pizka and Thomas Panas. Establishing economic effectiveness through software health-management. Technical report, Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States), 2009.

[5] Markus Schnappinger, Mohd Hafeez Osman, Alexander Pretschner, Markus Pizka, and Arnaud Fietzke. Software quality assessment in practice: a hypothesis-driven framework. In Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, page 40. ACM, 2018.

[6] Veda C Storey. Comparing relationships in conceptual modeling: mapping to semantic classifications. IEEE Transactions on Knowledge and Data Engineering, 17(11):1478–1489, 2005.

[7] Matthew West. Developing high quality data models volume 1: Principles and techniques. The Data Management Guide, 1994.

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@tum.de. Your email should explain your interest in the topic and contain your current transcript of records. The most promising candidates will be invited for an informal interview. Upon mutual agreement, the thesis can be sponsored by itestra GmbH.