Ciência de Dados e I.A.
Escola de Matemática Aplicada
Fundação Getúlio Vargas

Engenharia de Requisitos

TCC

Enhancing Model-Driven Reverse Engineering Using Machine Learning

Aluno: Isabela Yabe
Orientador: Rafael de Pinho André
Escola de Matemática Aplicada, FGV/EMAp
Rio de Janeiro - RJ.

Rio de Janeiro, 2025

Sumário

1 Revisão literária

Artigo revisado Siala (2024):

A revisão tem o objetivo de compreender o estado da arte das abordagens de engenharia reversa que partem de código-fonte e produzem artefatos de alto nível, como diagramas UML. Para garantir uma análise sistemática e comparável entre diferentes propostas, foram definidas perguntas de pesquisa (*Research Questions — RQs*) que orientam a coleta e síntese dos dados extraídos dos estudos selecionados.

- **RQ1.** Em quais linguagens e domínios as abordagens que partem de código-fonte foram aplicadas?
- **RQ2.** Quais modelos/artefatos de alto nível são gerados?
- **RQ3.** Qual aspecto é privilegiado (estático, dinâmico, híbrido) e com qual objetivo (compreensão, redocumentação, migração, qualidade)?
- **RQ4.** Quais técnicas e transformações viabilizam a passagem do código para o modelo de alto nível?
- **RQ5.** Quais ferramentas/frameworks são utilizados?
- **RQ6.** Como as abordagens são validadas e com que qualidade prática?

2 Em quais linguagens e domínios as abordagens que partem de código-fonte foram aplicadas?

Java e Python são as linguagens de código-fonte alvo do método MDRE proposto. O artigo situa o problema em sistemas legados corporativos e industriais, caracterizados por alta complexidade e longa vida útil.

3 Quais modelos/artefatos de alto nível são gerados?

O método MDRE proposto visa gerar especificações UML (diagramas de classe) e OCL(restrições, invariantes e regras de negócio) a partir do código-fonte.

4 Qual aspecto é privilegiado (estático, dinâmico, híbrido) e com qual objetivo (compreensão, redocumentação, migração, qualidade)?

O aspecto privilegiado é estático, e o objetivo principal é compreensão e redocumentação, com foco secundário em migração de sistemas legados.

5 Quais técnicas e transformações viabilizam a passagem do código para o modelo de alto nível?

Segundo Siala (2024), a pesquisa busca utilizar LLMs para abstrair especificações UML/OCL a partir de código-fonte, questionando também quais informações devem ser coletadas por meio desses modelos para facilitar tal abstração e de que forma.

artigo, as técnicas e transformações que viabilizam a passagem do código-fonte para modelos UML/OCL de alto nível são estruturadas em um pipeline MDRE híbrido que combina transformações modelo–modelo (M2M) com técnicas de aprendizado de máquina (LLMs).

a transformação ocorre por meio de quatro etapas principais, cada uma correspondendo a um tipo de técnica aplicada:

- Pré-processamento: a tokenização e a simplificação semântica de código, prepara o código para interpretação por LLMs, reduzindo a complexidade sintática, atuando como um modelo intermediário (T2M).
- Codificação com LLMs: as técnicas de transformer-based encoder-decoder traduz o código em uma representação textual semântica (pré-UML/OCL). Análoga a uma transformação M2M semântica, onde o modelo intermediário é gerado a partir da codificação contextual feita pelo LLM.
- Pós-processamento e Model Repair: a correção automática e o refinamento sintático dos modelos gerados, faz garantir a consistência e completude das restrições OCL e dos elementos UML. Aplicando correções na representação textual.
- Geração de diagramas: transformação de texto para grafo UML (via PlantUML, Graphviz, Modelio). Representando a transformação M2V. Função: converter as representações textuais validadas em diagramas formais.

O artigo propõe um processo de aprendizado bidirecional, em que o modelo aprende tanto a traduzir de código para modelo, quanto de modelo para código, reforçando a consistência entre os domínios.

O método emprega uma pipeline de transformações híbridas, em que técnicas de análise estática, transformações modelo–modelo (MDA) e LLMs.

6 Quais ferramentas/frameworks são utilizados?

O artigo prevê a geração automática de diagramas UML (principalmente de classes) a partir das saídas textuais do modelo, o artigo cita: Graphviz, PlantUML, Modelio.

Utiliza o frameworks de modelagem e padronização (OMG/MDA, AgileUML) e incorpora LLMs para aprendizado semântico.

O trabalho de Siala (2024) propõe um ecossistema integrado de ferramentas e frameworks que combinam:

- o paradigma MDA (para estrutura e interoperabilidade de modelos);

- LLMs baseados em Transformer (para tradução semântica de código),
- e ferramentas UML consagradas (Graphviz, PlantUML, Modelio, AgileUML) para gerar automaticamente modelos UML e OCL a partir de código-fonte Java e Python, com vistas à compreensão, redocumentação e migração de sistemas legados.

7 Como as abordagens são validadas e com que qualidade prática?

A validação da abordagem é planejada de forma estruturada dentro do paradigma de Design Science Methodology (DSM) e envolve comparação empírica, estudos de caso e critérios objetivos de qualidade de modelo.

A validação prática será realizada em duas etapas: (i) Comparação com baseline: aplicar a mesma entrada (código-fonte) em um método MDRE tradicional e na nova abordagem baseada em LLM; (ii) Estudos de caso: dois sistemas de software reais, de diferentes tamanhos e domínios.

Os critérios de avaliação para engenharia reversa incluem (i) a correção semântica e a completude dos modelos em comparação com o código; (ii) a qualidade e a compreensibilidade dos diagramas.

Autores / Referência	**Linguagem / Domínio**	**Modelo Gerado**	**Aspecto**	**Técnica / Transformação**	**Ferramenta / Framework**	**Validação / Estudo de Caso**
Siala (2024)	Java; Python; sistemas legados	UML Classe; OCL	Estático — compreensão, redocumentação e migração	código → tokenização/simplificação → geração textual UML/OCL intermediária(LLM) → model repair → diagramas UML/OCL	Graphviz; PlantUML; Modelio; AgileUML; LLM	Comparação MDRE: dois estudos de caso; correção semântica, completude e compreensibilidade

Tabela 1: Resumo das abordagens MDRE baseadas em código-fonte

Referências

SIALA, H. A. Enhancing Model-Driven Reverse Engineering Using Machine Learning. Versão inglesa. *In:* 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion). Lisbon, Portugal: IEEE/ACM, 2024. p. 1–13. King's College London, London, UK.