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

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 RQ1. 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 RQ2. Quais modelos/artefatos de alto nível são gerados?

O método MDRE proposto tem como objetivo gerar especificações UML (diagramas de classe) e OCL (restrições, invariantes e regras de negócio) a partir de código-fonte. Esses artefatos representam tanto a estrutura quanto a semântica dos sistemas, permitindo a compreensão e redocumentação automatizada.

4 RQ3. 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. A abordagem propõe combinar análise estrutural com abstração semântica por meio de LLMs, o que amplia o potencial de entendimento do código.

5 RQ4. Quais técnicas e transformações viabilizam a passagem do código para o modelo de alto nível?

A proposta estrutura-se como um pipeline MDRE híbrido que integra transformações modelo–modelo (M2M) tradicionais com técnicas de aprendizado de máquina baseadas em LLMs. O fluxo é dividido em quatro etapas principais:

- **Pré-processamento:** tokenização e simplificação semântica do código, reduzindo a complexidade sintática e preparando-o para interpretação pelos LLMs. Essa fase atua como um modelo intermediário (*Text-to-Model*).
- **Codificação com LLMs:** uso de modelos *transformer*-based *encoder-decoder* para traduzir o código em uma representação textual semântica intermediária (pré-UML/OCL), análoga a uma transformação M2M semântica.
- **Pós-processamento e *Model Repair*:** refinamento sintático e correção automática dos modelos gerados, assegurando consistência e completude das restrições OCL e dos elementos UML.
- **Geração de diagramas:** conversão das representações textuais validadas em diagramas formais UML e OCL, utilizando ferramentas como PlantUML, Graphviz e Modelio (fase M2V).

O processo adota um aprendizado bidirecional, no qual o modelo aprende a traduzir tanto de código para modelo quanto de modelo para código, reforçando a consistência entre os domínios e aprimorando a capacidade de generalização.

6 RQ5. Quais ferramentas/frameworks são utilizados?

A proposta combina o paradigma MDA para interoperabilidade de modelos com técnicas de aprendizado profundo e ferramentas UML consolidadas. O ecossistema integra:

- o framework **OMG/MDA** e a ferramenta **AgileUML** para padronização e estruturação dos modelos;
- modelos de linguagem de grande escala (**LLMs**) baseados em arquitetura Transformer, responsáveis pela codificação e abstração semântica;
- ferramentas de modelagem **Graphviz**, **PlantUML** e **Modelio**, utilizadas para gerar representações visuais UML e OCL a partir das saídas textuais.

Essa combinação sustenta um pipeline MDRE automatizado voltado à compreensão, redocumentação e migração de sistemas legados escritos em Java e Python.

7 RQ6. 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.