Summary of "An Empirical Study on the Potential Usefulness of Domain Models for Completeness Checking of Requirements"

Introduction

The completeness of requirements is a crucial yet challenging aspect of requirements validation. Requirements completeness can be categorized into internal and external completeness. Internal completeness ensures that a specification is self-contained, while external completeness ensures that all necessary system information is captured. Since external completeness cannot be guaranteed in absolute terms, domain modeling has been suggested as a method for improving it.

A domain model is a structured representation of the key concepts and their relationships within a given application domain. It can be used to cross-check natural-language (NL) requirements for missing or underspecified elements. This study empirically evaluates the effectiveness of domain models, represented as UML class diagrams, in detecting omissions in functional requirements expressed as "shall" statements.

Research Question and Approach

The central research question is: "Are domain models represented as UML class diagrams useful for checking the completeness of functional requirements expressed as shall statements?" The authors conduct an empirical study through case study research in three industrial domains. Experts construct domain models for these domains, and the models' sensitivity to omissions in requirements is analyzed through a randomized simulation process. Sensitivity refers to whether a domain model contains information that can help identify missing or under-specified requirements.

Methodology

The study includes three case studies from different industrial domains:

- Case A: Aerospace simulator module (163 functional requirements)
- Case B: Cyber-physical sensor platform (212 functional requirements)
- Case C: Content management system for safety assurance (110 functional requirements)

The research process involves three main steps:

1. **Domain Model Construction:** Subject-matter experts create domain models for each case study, ensuring the models are as complete as possible.
2. **Traceability Mapping:** Each requirement is mapped to relevant domain model elements.
3. **Omission Simulation and Sensitivity Analysis:** Requirements and their components are systematically removed, and the impact on the domain model is measured to determine sensitivity.

The omissions are categorized into:

- **Unspecified requirements** (entire requirements missing)
- **Under-specified requirements** (missing details such as constraints, conditions, or objects)

Key Findings

1. Sensitivity of Domain Models to Omissions

- Domain models exhibit **near-linear sensitivity** to missing or under-specified requirements.

- The sensitivity is significantly higher (on average **4.4 times more**) for unspecified requirements than for under-specified ones.
- Concepts in domain models tend to appear multiple times in requirements, reducing the model's sensitivity to small omissions.

2. Relationship Between Sensitivity and Requirements Characteristics

- Case C showed lower sensitivity than Cases A and B because domain concepts appeared more frequently in the requirements.
- The frequency of domain model elements appearing in requirements affects how easily omissions can be detected.

3. Use of Keyphrases as a Surrogate for Sensitivity Prediction

- Instead of relying on domain model elements, sensitivity can be approximated using unsupported keyphrases (noun phrases and verb phrases) after an omission.
- Strong correlation ($r > 0.96$, $p < 0.0001$) was found between unsupported keyphrases and unsupported domain model elements.
- This method allows practitioners to estimate sensitivity without building a domain model.

Conclusion and Future Work

The study provides empirical evidence that domain models can help identify missing and under-specified requirements, though their effectiveness depends on how frequently concepts are referenced in the requirements. The results suggest that domain models should be complemented by other techniques for completeness checking. Future work should focus on user studies to evaluate whether analysts can effectively leverage domain models in practice.