AI4SE: Semi-Automated Development of Textual Requirements: Combined Natural Language Processing and Multi-Domain Semantic Approach

By
UMD Team: Mark Austin, Sachraa Borjigin, Edward Zontek-Carney
Early Contributors: Andres Arellano, Leonard Petnga
September 21-22, 2022

AI4SE/SE4AI Workshop

www.sercuarc.org

Cleared for Public Release
Motivation: Formal Modeling of Requirements

Modern Engineering Systems

Nearly always designed, built, operated and maintained by teams of people + automation. Key component in bringing this capability together is the ability to write and manage textual requirements early in the system development lifecycle.

State-of-the-Art Practice

Key Problems

- Poor writing skills; excessive use of acronyms.
- Manual translation process is slow and error-prone.
- How do we know if a system description is complete and consistent?
Motivation: Formal Modeling of Requirements

Recent Advances

- Increased adoption of SysML in the profession – semi-formal representations enhance quality of communication among team members, reduce uncertainties and improve project schedule.
- Remarkable advances in AI (semantic modeling and reasoning) and ML.
- Use of text persists (e.g., for standards, regulations, etc).

Formal Representation of Requirements

- Correct (syntax and semantics) encoding of textual requirements.
- Automatic processing and traceability in the design process.
- Specific, verifiable, realistic, time-bound.

**Req:** Within the *school area*, maximum *speed* of *vehicles* shall be *15 mph* when children are present.
Motivation: Formal Modeling of Requirements

Brainstorming Project Ideas (Fall Semester, 2013)

Idea 1 (Syntax): Use knowledge in semantic graphs (ontologies and rules) to write template-compliant textual requirements.

Idea 2 (Semantics): Support writing (development) of semantically compliant requirements.

Idea 3 (Integration): Provide automated support for verification of requirements at various levels of design.

So how might this work?
Research Questions

- How can multi-domain semantic modeling and natural language processing work together to improve the development of textual requirements?
- How can requirements templates improve quality of validation of individual requirements and groups of requirements?

Motivating Case Study: New Computer Science Building at UMD
Sample Textual Requirements

Building Construction Requirements

Project Cost

1.1. The construction project budget shall be no more than US $150 million dollars.

Project Schedule

2.1. The construction project start date shall be on January 20, 2016.
2.2. The construction project end date shall be on April 20, 2019.
2.3. The construction project duration shall not exceed 4 years.

Building Foundation

3.1. The building foundation shall be constructed of solid materials.
3.2. The building foundation depth shall be at least 5 m.
3.3. The foundation wall thickness shall be no less than 15 cm.
3.4. The clear distance of building foundations to the construction site boundary shall be at least 60 cm.
Sample Textual Requirements

Building Construction Requirements

**Construction Site**

1. The construction manager is required to develop a project-specific orientation for all workers.
2. The general contractor shall have a process of validating training and certification in place for all construction workers.

**Construction Equipment**

1. The general contractor shall maintain documentation of all equipment inspection.
2. The crane operations shall be suspended if the wind speed is greater than 40 km/hr.

**Construction Workers**

1. The construction workers shall remain a minimum of 6 m distance from any solid red labeled exhaust system in the construction site.
Proposed Approach (What’s New?)

Textual Requirement Sentences → Syntax Tree → Template Matching → Validation

Feedback

Pages of Text → Import → NLP

chunking grammar → chunking grammar

Requirements Templates → Template Matching → Property Analysis and Validation

System Properties → System Ontology
Subsystem Properties → Subsystem Ontologies
Component Properties → Component Ontologies
Time Properties → Time Ontology
Physical Properties → Physical Ontology

Domain Ontologies

Meta-Domain Ontologies

Scope of FLOOR Textual Requirements
Software Prototype.

Proposed Approach (What’s New?)

Legend for Graphical Notation:
- Round rectangles for classes and primitive data types.
- Boxes represent software modules and individuals in semantic model.
- Solid and dashed one-way arrows represent data property and object property relationships.

Source: Borjigin, Austin, and Zontek-Carney, 2022
Multi-Domain Semantic Modeling

Synthesis of System Behavior and System Structure

Guiding Principles

- Data-driven approach.
- Co-development of ontologies, rules and data models.
- Ontologies visit data models to get individuals.
- Enhance power of rules with backend software functions.
- Semantic graph dynamically responds to incoming events.
Multi-Domain Semantic Modeling

Data-Ontology-Rule Footing (UMD / NIST / SERC in 2017).

Multi-domain Semantic Modeling

Domain A
- Rules A
- Ontology A
- Data Source A

Domain B
- Rules B
- Ontology B
- Data Source B

Executable Processing of Events
- Rules Engine
- Semantic Graph
- Events !!!

Source: Coelho, Austin, Blackburn, 2019
Multi-Domain Semantic Modeling

Executable Processing of Events

- **Requirements**
  - Requirement.rules
  - Crane.rules
  - Weather.rules
  - Geo-spatial.rules

- **Equipment**
  - Crane.owl
  - Weather.owl
  - Geo-spatial.owl

- **Environment**
  - Crane Data Model
  - Weather data model
  - Geo-spatial data model

- **Framework for Executable Processing of Events**
  - Reasoner
  - Semantic Graphs
  - Graph transformation

- **Meta Domain Ontologies and Rules**
  - Temporal Ontology
  - Physical Units Ontology
  - Spatial Ontology
  - Temporal Rules
  - Physical Units Rules
  - Spatial Rules

- **Building Footprint**
- **Site Boundary**
Working with Apache Jena and Jena Rules

Data-Driven Approach to Generation of Individuals in Semantic Graphs

```
<< abstract >>
AbstractOntologyModel

Jena Semantic Model

Jena Rules

Ontology

Building Domain Data Model

load

visit

hosting visitor

extend

XML Data File

load

load

real world building environment

Site Boundary
Building Footprint
building construction data model
external software

fact 1 built-in function fact 2 … fact 3

AND

derived fact 4

add new assertion to semantic model

Forward Chaining of Facts and Results of Built-in Functions to New Assertions…

Data-Driven Approach to Generation of Individuals in Semantic Graphs
Meta-Domain Ontologies and Rules

Modeling Time

In order for a decision involving time to be reliable, the underlying models of time and theories of reasoning need to be formal.

Temporal Domain

(a) Ontology for Instants and Intervals of Time

(b) Logical Relationships Among Intervals of Time

Relation | Inverse
---|---
Before(i,j) | After(j,i)
Meets(i,j) | MetBy(j,i)
Overlaps(i,j) | OverlappedBy(j,i)
Starts(i,j) | StartedBy(j,i)
During(i,j) | Contains(j,i)
Finishes(i,j) | FinishedBy(j,i)
Equals(i,j) | Equals(j,i)

Source: Allen, 1983.
Meta-Domain Ontologies and Rules

Physical Units Domain: Length, Mass, time, Temperature

\[ \text{unit} = k \cdot L^\alpha M^\beta t^\gamma T^\delta \cdot \text{rad}^\varepsilon \]

Semantic Representation for Physical Quantities and Units
Domain-Specific Ontologies and Rules

Requirement, Constraint, Template, and Task Ontologies
Framework for Linking Ontology Objects and Textual Requirements.
Case I: Detailed Validation of an Individual Requirement

Requirement: 5.2

The crane operations shall be suspended if the wind speed is greater than 40 km/hr.
Case 2: Identifying Requirement Duplicates

Requirement: 3.2

- **Req (I):** The building foundation depth shall be at least 500 cm.
  - **Constraint (I):** The building foundation depth shall be at least 5 m.
  - **isCompatibleWith:** Req (I) and Req (II)
  - **isVerifiedBy:** Req (I)
  - **hasDescription:** Constraint (I)

- **Req (II):** The building foundation depth shall be at least 5 m.
  - **Constraint (II):** The building foundation depth shall be at least 500 cm.
  - **isVerifiedBy:** Req (II)
  - **hasDescription:** Constraint (II)
Case 3: Group of Requirements Relating to a Single Task

Requirements: 2.1, 2.2 and 2.3

Schedule Requirements:

- **Req (I):** The construction project start date shall be on January 20, 2016.
- **Req (II):** The construction project end date shall be on April 20, 2019.
- **Req (III):** The construction project duration shall not exceed 4 years.

Constraint:

- Construction hasTask Consistency Check

Task:

- Name: Construction Project
- Start date: January 20, 2019
- End date: December 20, 2020
- Duration: not exceed 3 years

"Task" Consistency Check:

Feedback on "task" consistency check


Questions?

Contact Information

Mark Austin: austin@umd.edu, http://www.isr.umd.edu/~austin