Finding Flaws in Natural Language Requirements

This installment of Computer’s series highlighting the work published in IEEE Computer Society journals comes from IEEE Transactions on Software Engineering.

Matthew B. Dwyer, University of Nebraska–Lincoln

A software system begins in a user’s mind. With some coaxing, the system’s “idea” is expressed as natural language requirements. These requirements then drive the development of the software.

Existing approaches often structure natural language with templates that reduce ambiguity and normalize their format, making the requirements easier to understand and analyze. A key challenge is determining whether the natural language sentences conform to the template format. Prior work allowed conformance checking, given a glossary defining the problem-domain keywords. However, practitioners often write requirements as a means of discovering and finalizing the glossary, which is difficult to do in advance.

In “Automated Checking of Conformance to Requirements Templates Using Natural Language Processing” (IEEE Trans. Software Eng., vol. 41, no. 10, 2015, pp. 944–968), Chetan Arora and his colleagues explore whether modern approaches to natural language processing (NLP) can effectively find template conformance flaws in the kind of natural language requirements that practitioners create.

The authors describe the REquirements Template Analyzer (RETA), which uses NLP text-chunking techniques to overcome the need for a glossary. RETA, shown in Figure 1, integrates with existing requirements-authoring tools, enhancing them with the ability to report on conformance with different template structures, such as Chris Rupp’s and EARS (Easy Approach to Requirements Engineering). RETA can also detect problematic constructs in requirements; for example, the requirement “The S&T module shall process the query data and send a confirmation to the database” might be ambiguous because “and” implies temporal ordering.

The paper evaluates the proposed method on four case studies reflecting the real-world variability found in industrial requirements specifications. This evaluation’s breadth and depth effectively settles the question of how to check template conformance of natural language requirements for practicing engineers.

MATTHEW B. DWYER is a professor of computer science and engineering at the University of Nebraska–Lincoln. Contact him at dwyer@cse.unl.edu.