

Automated Conversion from a Requirements Document to an Executable Formal Specification

Beum-Seuk Lee

Department of Computer and Information Sciences
The University of Alabama at Birmingham
Birmingham, Alabama, U.S.A. 35294-1170
leebs@cis.uab.edu

Many formal specification languages have been developed to engineer complex systems. However natural language (NL) has remained the choice of domain experts to specify the system because formal specification languages are not easy to master. Therefore NL requirements documentation must be reinterpreted by software engineers into a formal specification language. When the system is very complicated, which is mostly the case when one chooses to use formal specification, this conversion is both non-trivial and error-prone, if not implausible. This challenge comes from many factors such as miscommunication between domain experts and engineers. However the major bottleneck of this conversion is from the inborn characteristic of ambiguity of NL and the different level of the formalism between the two domains of NL and the formal specification. This is why there have been very few attempts to automate the conversion from requirements documentation to a formal specification language.

This research project is developed as an application of formal specification and linguistic techniques to automate the conversion from a requirements document written in NL to a formal specification language. Contextual Natural Language Processing (CNLP) is used to handle the ambiguity problem in NL and Two Level Grammar (TLG) is used to deal with the different formalism level between NL and formal specification languages to achieve automated conversion from NL requirements documentation into a formal specification (in our case the Vienna Development Method - VDM++). First a knowledge base is built from the NL requirements documentation using CNLP by parsing the documentation and storing the syntactic, semantic, and contextual information. In this phase, the ambiguity is detected and resolved, if possible. Then the knowledge base is converted into

TLG by removing the contextual dependency in the knowledge base. Well structured and formalized data representations especially for the context are used to make smooth translations from NL requirements into the knowledge base and then from the knowledge base into a TLG specification. Due to its NL-like syntax and flexibility without losing its formalism, TLG is chosen as a formal specification to fill the gap between the different level of formalisms of NL and formal specification language. Finally the TLG code is translated into VDM++ by data and function mappings. Once we have translated the TLG specification into a VDM++ specification we can convert this into a high level language such as JavaTM or C++, using the code generator that the VDM ToolkitTM provides. Not only is this code quite efficient, but it may be executed, thereby allowing a proxy execution of the requirements. Another advantage of this approach is that the VDM Toolkit also provides for translation into the Unified Modeling Language (UML) using a link with Rational RoseTM. Therefore by using the formalized context in CNLP and TLG as a bridge between the requirements document and a formal specification language, we can achieve an executable NL specification for a rapid prototyping and reusability of requirements, as well as development of a final implementation.

Acknowledgements: This material is based upon work supported by, or in part by, the U. S. Army Research Laboratory and the U. S. Army Research Office under contract/grant number DAAD19-00-1-0350 and by the U. S. Office of Naval Research under award number N00014-01-1-0746. The author thanks IFAD for providing an academic license to the IFAD VDM Toolbox in order to conduct this research.