Chapter 37
Fuzzy Ontology for Requirements Determination and Documentation During Software Development

Priti Srinivas Sajja
https://orcid.org/0000-0002-9676-0885
Sardar Patel University, India

Rajendra A. Akerkar
Western Norway Research Institute, Norway

ABSTRACT

Every business has an underlying information system. Quality and creditability of a system depend mainly on provided requirements. Good quality requirements of a system increase the degree of quality of the system. Hence, requirements determinations is of prime importance. Inadequate and misunderstood requirements are major problems in requirements determination. Major stakeholders of the requirements are non-computer professional users, who may provide imprecise, vague, and ambiguous requirements. Further, the system development process may be partly automated and based on platform such as web or Semantic Web. In this case, a proper ontology to represent requirements is needed. The chapter proposes a fuzzy RDF/XML-based ontology to document various requirements. A generic architecture of requirements management system is also provided. To demonstrate the presented approach, a case of student monitoring and learning is presented with sample software requirements specifications and interfaces to collect requirements. The chapter concludes with advantages, applications, and future enhancements.

DOI: 10.4018/978-1-6684-3702-5.ch037

INTRODUCTION

The quality of any software depends on the requirements considered during the development of the software. Requirements generally provide a basic skeleton of the software. The document containing well-formed requirements serve the basis for all phases of the software development activity. The inclusion of good quality requirements in the software requirement specifications leads towards good quality software. After proper analysis phase, once requirements are collected, analyzed and documented; a Software Requirements Specification (SRS) will be prepared. The SRS will be useful at the beginning of the design phase as well as at the end of the design phase to test whether the specified requirements are accommodated in the proposed design or not. Coding, testing and evaluation of the software are also done according to the requirements.

The requirements often contain imprecision and vagueness within them. Further, the importance of each requirement is different and affected by various parameters such as requirement initiator's (who has initiated the requirement) mindset, cost of adding the requirements, loss due to missing of the requirements, the priority of the requirements, etc. Such important but vague criteria can be added as a fuzzy tag to each requirement while documenting the requirements with the help of fuzzy logic. Fuzzy logic, with the virtue of fuzzy membership function, can efficiently handle such vagueness and impression in computer systems. In this scenario, there is a need for a documentation ontology that documents requirements on the Web platform and manages the fuzziness associated with it. In contrast to traditional knowledge-based approaches, e.g. formal specification languages, ontologies seem to be well suited for an evolutionary approach to the specification of requirements and domain knowledge (Wouters, Deridder, & Van Paesschen, 2000). Moreover, ontologies can be used to support requirements management and traceability.

Besides, varying requirements and evolving solutions are important challenges during the software development process. Agile software development is the way to tackle these challenges by adopting methods based on iterative and incremental development. The challenges are similar in the area of ontology engineering. Several situations ontology development is a continuous and collaborative task.

The proposed chapter introduces the current scenario and sets the necessary technical background of ontology, knowledge engineering and fuzzy logic in section 1 and section 2. After that, the chapter documents related work in the area of ontology, fuzzy logic, and use of ontology in software development activities with general observations and limitations. The related work is documented in section 3 of the chapter. The section also summarizes the survey on work done by presenting the observations and characteristics. Section 4 of the chapter proposes a fuzzy ontology for requirements determination. The section introduces various components of requirements with the necessary description along with the graphical representation of the components to highlight the relationship between them. An RDF/XML structure is proposed for the requirements documentation in section 4. A generic architecture to manage the fuzzy ontology repository along with a knowledge base and other components are also illustrated here. Section 5 discusses a case of a student's learning and monitoring system and presents sample software requirements specification with the requirements documented in the RDF/XML format and an interface screen for the acquisition of requirements. Section 5 also presents the fuzzy membership functions used for the experimental system. Section 6 presents advantages, applications and future directions based on the proposed approach.