Chapter 53
Natural Language Processing:
An Inevitable Step in Requirements Engineering

A. Egemen Yilmaz
Ankara University, Turkey

ABSTRACT

Requirement analysis is the very first and crucial step in the software development processes. On the other hand, as previously addressed by other researchers, it is the Achilles' heel of the whole process since the requirements lie on the problem space, whereas other software artifacts are on the solution space. Stating the requirements in a clear manner eases the following steps in the process as well as reducing the number of potential errors. In this paper, techniques for the improvement of the requirements expressed in the natural language are revisited. These techniques try to check the requirement quality attributes via lexical and syntactic analysis methods sometimes with generic, and sometimes domain and application specific knowledge bases.

INTRODUCTION

System/software requirement quality is one of the most important drivers for the success of a final system/software related product. But ironically, as stated by Kof (2005), in practice "requirements engineering is the Achilles' heel of the whole software development process", because requirements documents are usually inconsistent and incomplete. Cheng and Atlee (2007) identify the main reason for this as follows: "… because requirements reside primarily in the problem space[;] whereas[,] other software artifacts reside primarily in the solution space".

A study showed that the user involvement is the most important factor for the success of the software development projects (Standish Group, 1995). Moreover, even in the cases where the user involvement is sufficient, the success is dependent on clear statement of the requirements, which appears as the third most important factor on the list. In addition, other studies showed that the underlying reason for 60 to 85% of the software errors during a system's life time is nothing but the requirement defects (Davis, 1990; Schach, 1992; Young, 2001). On the other hand, if such defects could not be fixed at the early

phases of the projects, the cost of fixing the error would dramatically increases at each phase of the development life cycle as seen in Table 1 (Young, 2001). Defects detected during requirement analysis and design phases could reduce the rework effort between 40% and 50% (Boehm, 1981; Boehm & Basili, 2001; Gause & Weinberg, 1989).

All the above studies support that, effective customer interaction and defect-free requirement engineering processes are the key factors for successful system/software development.

Pohl's requirement engineering process model (Pohl, 1994), depicted in Figure 1, might be considered as a reference for the improvement the software requirement quality. Pohl's model suggests focusing on the three dimensions seen in Figure 1. Stakeholders of the requirement engineering process should perform a progress on the specification, representation and agreement dimensions. Hence, tools supporting progress in any of those three dimensions would increase the success rates of software development projects.

Table 1. Relative cost of fixing an error (Young, 2001)

Phase in which the Error is Found	Relative Cost
Requirements Analysis	1
Design	3-6
Coding & Unit Test	10
Development Testing	15-40
Acceptance Testing	30-70
Operation	40-1000

Figure 1. Pohl's requirement engineering model (Adapted from (Pohl, 1994))
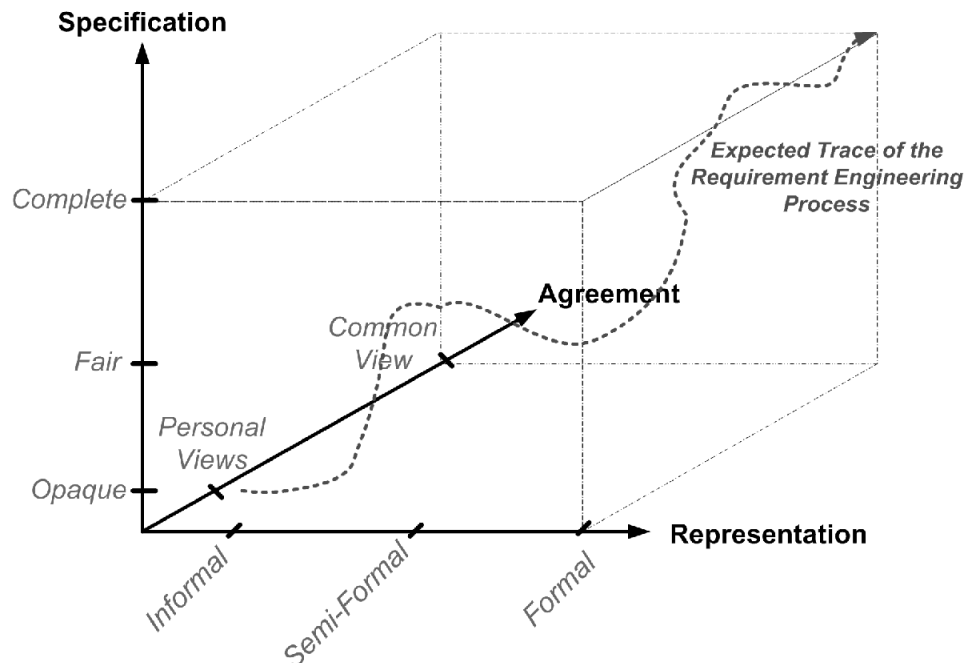
15 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/natural-language-processing/173382

Related Content

A Novel Sparse Representation Based Visual Tracking Method for Dynamic Overhead Cranes: Visual Tracking Method for Dynamic Overhead Cranes
Tianlei Wang, Nanlin Tan, Chi Zhang, Ye Liand Yikui Zhai (2019). *International Journal of Ambient Computing and Intelligence (pp. 45-59).*
www.irma-international.org/article/a-novel-sparse-representation-based-visual-tracking-method-for-dynamic-overhead-cranes/238053

Exploring the Role of AI-Driven Tools in Evaluating Pedagogical Competencies
Rajashree Das, Durga Prasad Singh Samanta, Gouranga Nandaand Minakshi Rani Mohanty (2024). *Facilitating Global Collaboration and Knowledge Sharing in Higher Education With Generative AI (pp. 89-108).*
www.irma-international.org/chapter/exploring-the-role-of-ai-driven-tools-in-evaluating-pedagogical-competencies/336033

Eliciting User Preferences in Multi-Agent Meeting Scheduling Problem
Mohammad Amin Rigiand Farid Khoshalhan (2011). *International Journal of Intelligent Information Technologies (pp. 45-62).*
www.irma-international.org/article/eliciting-user-preferences-multi-agent/54066

The Robot Wrote My College Papers: Integrating Chatbots to Assist Higher Education
Ivy Shen (2024). *Generative AI in Teaching and Learning (pp. 311-327).*
www.irma-international.org/chapter/the-robot-wrote-my-college-papers/334784

Finding "H" in HRI: Examining Human Personality Traits, Robotic Anthropomorphism, and Robot Likeability in Human-Robot Interaction
Anshu Saxena Arora, Mayumi Fleming, Amit Arora, Vas Tarasand Jiajun Xu (2021). *International Journal of Intelligent Information Technologies (pp. 1-20).*
www.irma-international.org/article/finding-h-in-hri/272006