

## Chapter 3.3

# Natural Language Processing Techniques in Requirements Engineering

**A. Egemem Yilmaz**

*Ankara University Faculty of Engineering, Turkey*

**I. Berk Yilmaz**

*Ankara University Faculty of Engineering, Turkey*

### **ABSTRACT**

*Requirement analysis is the very first and crucial step in the software development processes. Stating the requirements in a clear manner, not only eases the following steps in the process, but also reduces the number of potential errors. In this chapter, 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. 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, another study showed that the underlying reason for 85% of the software errors is nothing but the requirement defects (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 increase at each phase of the development life cycle as seen in Table 1 (Young, 2001). Defects detected dur-

DOI: 10.4018/978-1-61350-456-7.ch3.3

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

ing 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

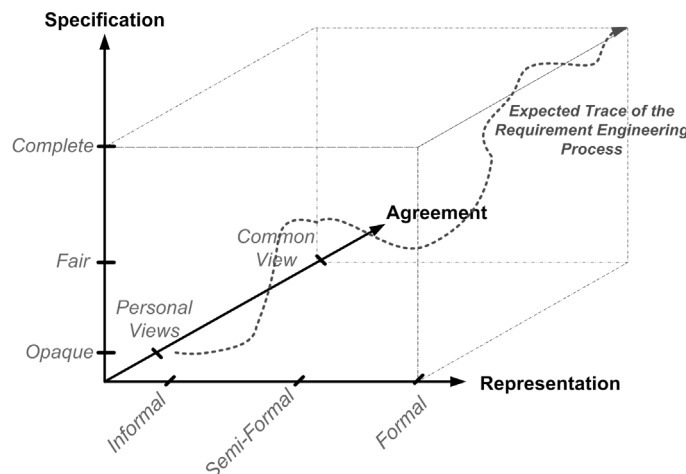
dimensions. Hence, tools supporting progress in any of those three dimensions would increase the success rates of software development projects.

The applicability of natural language processing techniques is best understood via the following two-phase requirement engineering model. Figure 2 depicts a high level representation of these two stages.

The first phase, which can be referred to as “Perception of the domain”, is a set of activities performed in order to understand the domain components and their relationships. Engineers embark upon to this stage at the very early phases of the development. This cognitive stage is usually followed by the representation of the information learned so far, where this representation is required to guarantee the correctness and completeness.

The second phase includes the “Expression of the requirements” in terms of visual and verbal statements. Those statements are used for understanding the rest of the domain; they are also important for achieving a better understanding among the end user, the buyer (if different than the end user), and the development team(s). Both the specification and representation qualities are important in order to deliver the right ‘message(s)’

Figure 1. Pohl’s requirement engineering model (Adapted from Pohl, 1994)



11 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-techniques-requirements/62463](http://www.igi-global.com/chapter/natural-language-processing-techniques-requirements/62463)

## Related Content

---

### DNA Computing

Tao Song, Xun Wang, Shudong Wang and Yun Jiang (2011). *Kansei Engineering and Soft Computing: Theory and Practice* (pp. 85-110).

[www.irma-international.org/chapter/dna-computing/46393](http://www.irma-international.org/chapter/dna-computing/46393)

### MEMS-Based Sensor Technologies: Bibliometric Trends, Simulation Analysis, and System-Level Integration

Aditya Shrivastava and Susanta Das (2026). *AI-Driven Hardware Security: Architectures, Chips, and Trust* (pp. 69-112).

[www.irma-international.org/chapter/mems-based-sensor-technologies/406398](http://www.irma-international.org/chapter/mems-based-sensor-technologies/406398)

### A Proposed Pragmatic Software Development Process Model

Sanjay Misra, M. Omorodion, Amit Mishra and Luis Fernandez (2018). *Computer Systems and Software Engineering: Concepts, Methodologies, Tools, and Applications* (pp. 607-622).

[www.irma-international.org/chapter/a-proposed-pragmatic-software-development-process-model/192895](http://www.irma-international.org/chapter/a-proposed-pragmatic-software-development-process-model/192895)

### Petri Net Based Deadlock Prevention Approach for Flexible Manufacturing Systems

Chunfu Zhong and Zhiwu Li (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 445-463).

[www.irma-international.org/chapter/petri-net-based-deadlock-prevention/62458](http://www.irma-international.org/chapter/petri-net-based-deadlock-prevention/62458)

### Girls and Computers - Yes We Can!: A Case Study on Improving Female Computer Confidence and Decreasing Gender Inequity in Computer Science with an Informal, Female Learning Community

Misook Heo and L. Monique Spradley-Myrick (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 1126-1143).

[www.irma-international.org/chapter/girls-computers-yes-can/62502](http://www.irma-international.org/chapter/girls-computers-yes-can/62502)